A Pose Invariant Face Recognition system using Subspace Techniques

by

Mohammed Aleemuddin

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

In Partial Fulfillment of the Requirements
for the Degree

MASTER OF SCIENCE

IN

Telecommunication Engineering

KING FAHD UNIVERSITY
OF PETROLEUM AND MINERALS

Dhahran, Saudi Arabia

November 2004

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **Mohammed Aleemuddin** under the direction of his thesis advisor and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN TELECOMMUNICATION ENGINEERING.**

THESIS COMMITTEE

____
Dr. Deriche, M. (Chairman)

____
Dr. Zerguine, A. (Member)

____
Dr. Sheikh, A.U.H. (Member)

____
Dr. Jamil M. Bakhashwain
(Department Chairman)

____
Dr. Mohammad Abdulaziz Al – Ohali
(Dean of Graduate Studies)

____
Date

Dedicated to

My Parents

ACKNOWLEDGMENT

In the name of Allah, the Most Gracious and the Most Merciful

All praise and glory be to Allah, for giving me an opportunity to pursue the M.S. program at King Fahd University of Petroleum and Minerals. I am very thankful to KFUPM for the research facilities and the work environment provided to me. These facilities enabled me to enrich my knowledge and carry out my research.

My appreciation and heartfelt gratitude goes to my thesis adviser Dr. M. Deriche for his constant endeavor, guidance and the time he devoted for this research work. I also wish to thank the other members of my thesis committee Dr. Sheikh, A.U.H and Dr. Zerguine, A.

The greatest moral support I always carry with me is from my father. He always supported me with his love affection and advise, which kept me going. From the bottom of my heart I appreciate my family for the patience, the support and the encouragement they gave me in carrying out this research work while I was away from home.

And last but not the least I would like to acknowledge my friends Riyaz, Anees, Mazher, Bakhtiar, Zahed, Ameer, Rizwan, Farooq, Kashif, Hameed, Babar, Faisal, Muddasir, Imran, Basharat and all others. They created joyful environment being in KFUPM campus. My special thanks goes to (Red Hat's Linux) Fedora project for their operating system, which made thesis writing an enjoyable work.

"Our lord! Give us in this world that which is good and in the Hereafter that which is good, and save us from the torment of the Fire!",[al-Baqarah 2:201].

Contents

Acknowledgment **ii**

List of Figures **viii**

List of Tables **xii**

Abstract (English) **xiii**

Abstract (Arabic) **xiv**

1 Introduction **1**

1.1 Motivation 3

1.2 Problem Statement 7

1.3 Major Contributions of the Thesis 8

1.4 Thesis Outline 9

2 Background **10**

2.1 Introduction to Biometric Systems 11

2.2 A Review of Various Biometrics 14

2.2.1 Selecting the Right Biometric Technology 24

2.3 Multimodal Biometric Systems 27

2.4 Face Recognition Techniques 28

2.4.1 Different Approaches to Face Recognition 30

2.4.2 Template Based Face Recognition 31

2.4.3 Appearance Based Face Recognition 32

2.4.4 Hybrid Appearance based techniques 36

2.4.5 Model Based Face Recognition 37

2.4.6 Face Recognition using Neural Networks 40

2.5 Standard Databases 42

2.6 Chapter Summary 43

3 Linear Subspace Techniques **44**

3.1 Vector Representation of Images 46

3.2 Principal Component Analysis 49

3.2.1 Experimental Results 56

3.3 Linear Discriminant Analysis 57

3.3.1 Experimental Results 62

3.4 Independent Component Analysis 63

3.4.1 Experimental Results 69

3.5 Comparison of PCA, LDA and ICA 70

3.6 Chapter Summary 72

4 Performance Evaluation Techniques for Face Recognition Systems 73

4.1 Distance Measures 74

4.2 Effect of Cropping 81

4.3 Face Verification and the ROC 83

4.3.1 The Watch-list ROC 84

4.3.2 Experimental Details of the Watch-list ROC 85

4.3.3 Verification ROC 90

4.4 Chapter Summary 93

5 Pose Invariant Face Recognition 94

5.1 Introduction 94

5.2 The Proposed Algorithm 96

5.3 Pose Estimation using LDA 97

5.3.1 Experimental Results of Pose Estimation 100

5.4 View Specific Subspaces 103

5.5 Training the View Specific Databases 104

5.5.1 View-based LDA 105

5.5.2 View-based PCA 106

5.5.3 Traditional PCA and Traditional LDA 106

5.6 Experimental Results 109
5.7 The KFUPM Capstone Database 112
5.8 Pose Estimation for 3, 5, and 7 pose systems 114
5.9 Experiments using the View-based system for 3, 5, and 7 poses 116
5.10 Chapter Summary 117

6 Conclusion and Future Work 118

6.1 Conclusion 118
6.2 Proposed Future Research Directions 119

List of Figures

1.1 Total biometric revenues[1] 4

1.2 Biometrics market shares[1] 5

1.3 A subject in different poses 7

2.1 A typical biometric system 12

2.2 Details of a fingerprint [2] 15

2.3 Location of the iris [3] 18

2.4 Location of the iris and the retina [3] 18

2.5 A sample image of a retinal scan 19

2.6 A typical recognition hand geometry system [4] 20

2.7 On-line processing of signatures [5] 22

2.8 Pictorial representation of the DNA [6] 23

2.9 Matching accuracy using a combination of face and fingerprint[7]. . . . 28

2.10 Comparison of various biometric features based on MRTD [8] 29

2.11 Classification of face recognition methods 30

2.12 Multiview faces overlaid with labeled graphs [9] 38

2.13 The 3-D morphable face models [10] 39

3.1 Samples of different subjects [11] 47

3.2 Samples of one subject in different conditions [11] 47

3.3 Samples of test faces used in evaluation of the PCA and the LDA [11] 48

3.4 Mean face from the training database 48

3.5 The first six eigenfaces 55

3.6 The eigenvalue spectrum 56

3.7 Recognition accuracy using PCA 58

3.8 The first six LDA basis 60

3.9 The eigenvalue spectrum of between-class and within-class covariance matrices. 61

3.10 Recognition accuracy using LDA 62

3.11 The first six ICA basis 68

3.12 Recognition accuracy using ICA 69

3.13 Recognition accuracy using PCA, LDA, and ICA 71

4.1 Recognition accuracy using the standard euclidean distance 76

4.2 Recognition accuracy using the Mahalanobis distance 77

4.3 Recognition accuracy using the city block metric 78

4.4 Recognition accuracy using the Minkowski metric 79

4.5 Recognition accuracy using the cosine distance 80

4.6 Test faces cropped at 15% 81

4.7 Test faces cropped at 30% 81

4.8 Test faces cropped at 60% 82

4.9 Maximum recognition accuracy at various levels of cropping 82

4.10 Samples of faces from the AT&T database 86

4.11 Mean face from the AT&T database 86

4.12 The first six eigenfaces 87

4.13 Watch-list receiver operating characteristic curve. 89

4.14 Verification receiver operating characteristic curve. 92

5.1 A typical subject in different poses 95

5.2 Block diagram of the pose invariant subspace system 97

5.3 Sample images with additive noise (SNR=10dB) 102

5.4 Effect of noise on pose estimation 102

5.5 Training images of one person in different poses 104

5.6 Test faces of one person in 4 poses 105

5.7 Mean images of all faces in front, left, and right pose 106

5.8 Fisher faces trained for front faces 107

5.9 Fisher faces trained for left faces 107

5.10 Fisher faces trained for right faces 107

5.11 Fisher faces of traditional LDA 107

5.12 Eigen faces trained for front faces 108

5.13 Eigen faces trained for left faces 108

5.14 Eigen faces trained for right faces 108

5.15 Eigen faces of traditional PCA 108

5.16 Mean images of all faces in traditional PCA and traditional LDA . . 109

5.17 View-Based LDA vs traditional LDA 109

5.18 View-Based LDA vs traditional PCA 110

5.19 View-Based PCA vs traditional PCA 111

5.20 Snapshots of the 150 images generated from a video 113

List of Tables

2.1 Comparison of various biometric techniques [12]. 26

2.2 Comparison of different subspace techniques [8] 36

2.3 Recognition results of 3D morphable models [13] 40

2.4 List of available databases in the public domain [8] 42

3.1 Comparison various subspace algorithms for face recognition 71

5.1 Experimental results of pose estimation 100

5.2 Summary of results 111

5.3 Distribution of the captured images over the 7 poses intervals 114

5.4 Distribution of the captured images over the 5 poses intervals 115

5.5 Distribution of the captured images over the 3 poses intervals 115

5.6 Experimental results of pose estimation 115

5.7 Summary of experiments on recognition accuracy 116

THESIS ABSTRACT

Name: Mohammed Aleemuddin

Title: A Pose Invariant Face Recognition system using Subspace Techniques

Degree: MASTER OF SCIENCE

Major Field: Telecommunication Engineering

Date of Degree: November 2004

Several systems require authenticating a person before giving access to their resources. Biometrics has been long known to recognize persons based on their physical and behavioral characteristics. Face recognition is one such biometrics; it has received a considerable attention in recent years both from the industry and the research community. But face recognition is susceptible to variations in pose, light intensity, expression, etc. In this work, we propose to develop a robust pose invariant face recognition system using different linear subspace techniques.

Keywords*: Biometrics, Face recognition, Pose invariant.*

King Fahd University of Petroleum and Minerals, Dhahran.
November 2004

ملخص الرسالة

الاسم: محمد عليم الدين

عنوان الرسالة: تصميم نظام تعرف على الوجه من زوايا متعددة باستخدام طرق التجزيء

الدرجة: ماجستير

التخصص: هندسة الاتصالات

تاريخ الرسالة: نوفمبر 2004 م.

العديد من الأنظمة تتطلب التعرف على الأشخاص قبل إعطائهم الصلاحية للدخول أو الاطلاع على مقتنيات منظمة ما. علم التحليل الإحصائي للظواهر الحيوية استخدم منذ فترة طويلة لهذه الغاية وهي التعرف على الأشخاص من خلال خصائصهم الفيزيائية أو السلوكية. التعرف على الوجه هو أحد فروع هذا العلم التي حازت الكثير من الاهتمام في الآونة الأخيرة في المجالات البحثية والصناعية. غير أن هذا الفرع يحتوي على الكثير من العوائق التي قد تؤثر على كفاءة التعرف مثل تغير زاوية الوجه ، التغير في شدة الإضاءة، التغير في تعابير الوجه، الخ. هذه الرسالة تحتوي على تصميم وتطوير لنظام تعرف معتمد على الوجه من زوايا متعددة باستخدام طرق تجزيء خطية مختلفة.

كلمات مهمة: التحليل الإحصائي للظواهر الحيوية، التعرف على الوجه، متعدد الزوايا.

درجة الماجستير

جامعة الملك فهد للبترول والمعادن، الظهران

نوفمبر 2004

Chapter 1

Introduction

Humans have been using physical characteristics such as face, voice, gait, etc. to recognize each other for thousands of years. With new advances in technology, biometrics has become an emerging technology for recognizing individuals using their biological traits. This technology makes use of the fact that each person has specific unique physical traits that are one's characteristics which can't be lost, borrowed or stolen. By using biometrics it is possible to confirm or establish identity based on "who the individual is", rather than by "what the individual possesses" (e.g., an ID card) or "what the individual remembers" (e.g., a password).

Passwords determine identity through user knowledge, if someone knows the password, then that person can gain access to some restricted areas or resources of a certain system. The drawback is that a password has nothing to do with the actual person using it. Passwords can be stolen, and users can give their passwords to others, resulting in systems that are vulnerable to unauthorized people. There is no foolproof way to make password-protected systems safe from unauthorized users. There is no way for password-based systems to determine user identity beyond doubt. The initial intent of such schemes is, however, to ensure that the provided services are accessed only by an authorized user, and not anyone else.

Several systems require authenticating a person before giving access to their resources. Biometrics have been long known to recognize persons based on their physical and behavioral characteristics. Examples of different biometric systems include fingerprint recognition, face recognition, iris recognition, retina recognition, hand geometry, voice recognition, signature recognition, etc. Face recognition in particular, has received a considerable attention in recent years both from the industry and the research communities. The real-life challenge here is the identification of individuals in everyday settings, such as offices or living-rooms. The dynamic, noisy data involved in this type of task is very different to that used in typical computer vision research, where specific constraints are used to limit variations. Historically, such limitations have been essential in order to limit the computational burden required to process, store and analyze visual data. However, enormous improvements in computers in terms of speed of processing and size of storage media, accompanied by progress in statistical techniques, is making it possible to realize such complex systems.

1.1 Motivation

With the current advances in technology, biometrics is becoming part of day to day life, where a person is recognized by his/her personal biological characteristics. Biometrics enables a number of applications, which can be divided into the following three main groups: i.**Commercial** applications such as computer network login, electronic data security, e-commerce, Internet access, ATM, credit card, physical access control, cellular phone, PDA, medical records management, distance learning, etc., ii. **Government** applications such as national ID card, correctional facilities, driver's license, social security, welfare-disbursement, border control, passport control, etc., and iii. **Forensic** applications such as corpse identification, criminal investigation, terrorist identification, parenthood determination, missing children, etc.

The global industry revenues of $729m in year 2002 are expected to reach $1905.4m by 2005, shown in Figure 1.1, driven by large-scale public sector biometric deployments, the emergence of transactional revenue models, and the adoption of standardized biometric infrastructures and data formats.

Face recognition has received considerable interest as a widely accepted biometric, because of the ease in collecting face images of persons. Face recognition is being used in various applications like crowd surveillance, criminal identification, and criminal record, access to entry etc. Figure 1.2 shows that facial recognition

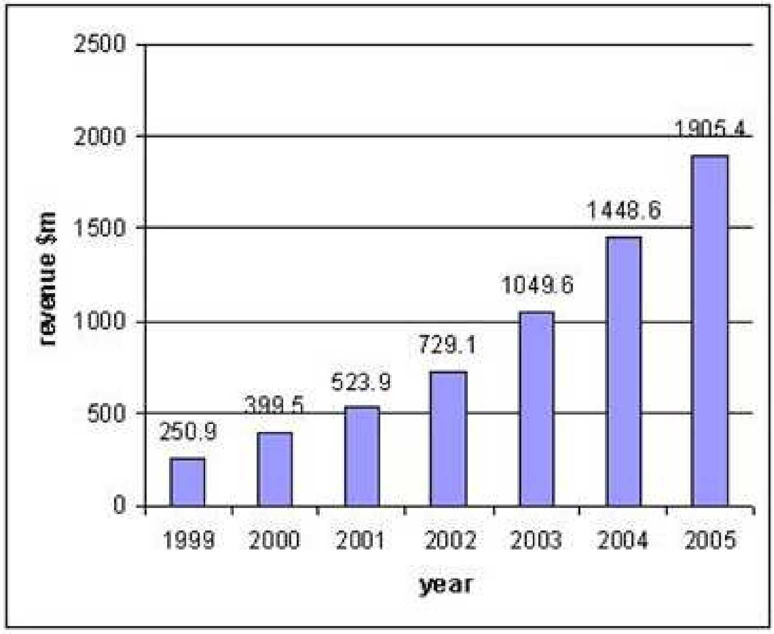

Figure 1.1: Total biometric revenues[1]

takes 15% of the total biometric market.

Face recognition developers, however, have to consider a number of major issues before face recognition systems become standard systems. The requirements for a useful, commercial face recognition and identity logging system, for small groups of known individuals in busy unconstrained environments, (such as domestic living-rooms or offices) can be split into several groups: i. General requirements that need to be satisfied by all components of the system, ii. Acquisitions requirements concerned with monitoring and extraction, iii. Face recognition requirements for the recognition stage, and iv. Identity requirements which are concerned with how the recognition output is used [2]. The acquisition, recognition, and identity requirements are detailed below:

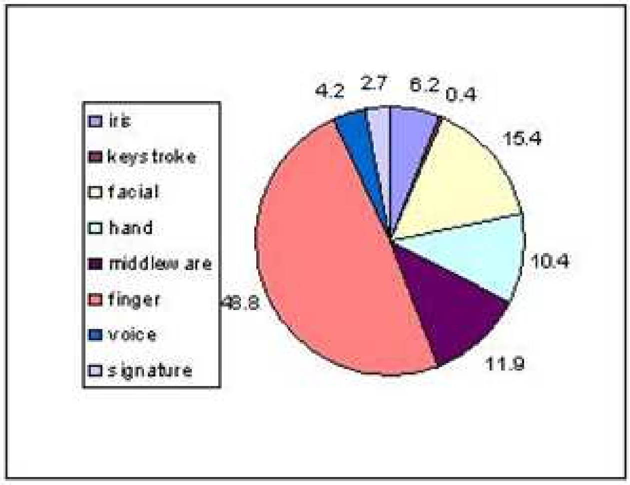

Figure 1.2: Biometrics market shares[1]

1. *Acquisition requirements:*

   - The major requirement is real-time tracking of individuals, with the ability to deal with multiple identities and occlusion. [1]

   - Real-time detection and localization of faces in acquired busy images.

2. *Face recognition requirements:*

   - Fast learning and real-time recognition of faces, with a minimum number of tunable parameters.

   - Ability to work with low resolution (size lesser then 50×50 pixels) face images, segmented from a single, wide angle view.

   - Invariance to typical variations in the environment, such as: a. variation

[1] Occlusion is defined as covering of face with objects like hat, facial hair, sun glasses, etc.

in shifted position and scale, b. lighting directions, c. occlusion by other objects, d. Variation in the image background areas of image.

- Invariance to typical facial variations, including: a. moderate expression variation, b. head pose orientation, c. day-to-day facial difference due to glasses, makeup, skin tones, etc, d. long-term facial changes (eg. aging).
- Level of confidence in the output available to be able to discard erratic or ambiguous data.
- Ability to detect, but not recognize, unknown individuals. (i.e. people from outside the "learned" group.)

3. *Identity Requirements:* Ability to adapt the known group of individuals (including information coming from a mechanism to handle 'strangers'), including

   - Learning details of a new individual.[2]
   - Forgetting a currently known individual.
   - Learning the new appearance of a currently known individual.
   - Identifying 'strangers'.

In designing robust face recognition systems it is primordial to consider the case of identification in dynamic, real-life environments as individuals may be standing or sitting and may look at the camera from different directions. This is what we

[2]Learning is defined as the process of extracting features from face image database using a face recognition algorithm.

call the problem of pose variation. Even if a person can be tracked and the head localized, the face could be facing in any direction. Consider the case of capturing a criminal's face; the criminal would make every attempt to avoid a biometric system recording his presence. Based on the challenges above, we formulated our problem as follows.

1.2 Problem Statement

The major difficulty in designing robust face recognition system is the wide variety in the input data. In particular, the problem is to recognize a subject with a varying pose, as shown in Figure 1.3. The problem of pose of the subject is one of the fundamental requirements for robust face recognition systems.

Figure 1.3: A subject in different poses

Pose variation is a non trivial problem to solve, as the pose of the face changes, new information is introduced which cannot be directly extracted from the different poses. There are various techniques proposed to overcome the problem of varying poses in face recognition. In earlier work, Growing Gaussian mixtures models [14] were proposed, but these require sufficient amount of training faces to be appropriate. An alternative to such models is the use of 3D face models [10]. However 3D models are expensive and difficult to develop. The view based eigenspaces developed by Moghaddam and Pentland [15] were also shown to outperform the single eigenspace approach.

In this research, we propose to extend the approach developed in [15] and apply it using Linear Discriminant Analysis (LDA). Contrary to the Principal Component Analysis, which is seen by most as a data reduction approach, LDA is better seen as a classification approach. In our experiments, we will show that View-based LDA performs better than View-based PCA. We will also demonstrate that LDA can be used to estimate the pose.

1.3 Major Contributions of the Thesis

The main contributions of the thesis are:

1. An extensive literature survey was carried on various biometrics with an aim to understand the techniques currently in use. The literature survey focused in particular on face recognition research.

2. We studied and implemented a number of linear subspace techniques, including PCA, LDA, ICA, and evaluated their performance using the Yale database.

3. We performed a number of experiments on various distance measures with an aim to select the best distance measure.

4. We developed a novel approach for face recognition which is invariant to pose. Our approach is a 2-stage algorithm using LDA. We showed that our algorithm outperforms all existing subspace approaches and is robust to cropping and additive noise.

1.4 Thesis Outline

The rest of the thesis is organized as follows: In Chapter 2, we present a brief overview of different face recognition techniques. In Chapter 3, we describe in details Linear Subspace Techniques, particularly PCA, LDA and ICA. We also present the results of recognition accuracy for experiments performed on the Yale database. Numerous experiments were then performed using various distance measures; these are described in Chapter 4. In Chapter 5, we discuss the problem of varying pose of a subject, and propose a new technique which uses multiple LDA subspaces. Finally in Chapter 6, we conclude the thesis with some future directions and a conclusion.

Chapter 2

Background

Several systems require authentication before giving access to certain resources. Traditionally, there have been three different types of authentication:

- *something you know:* a password, a PIN, or a piece of personal information (such as your mother's maiden name);
- *something you have:* a card key, a smart card, or a token (like a Secure ID card); and/or
- *something you are:* a Biometric!

Among these, a biometric is the most secure and convenient authentication tool existing to date. It can't be borrowed, stolen, or forgotten, and forging one is practically impossible (replacement part surgery, by the way, is not considered).

Biometrics measure the individual's unique physical or behavioral characteristics to recognize or authenticate identity.

2.1 Introduction to Biometric Systems

A complete biometric system comprises of both hardware and software; the hardware collects the data, and the software interprets the data and evaluates acceptability. The major steps in building an effective biometric system are as follows: (refer to Figure 2.1)

1. Capture the chosen biometric;
2. Process the biometric and extract and enroll the biometric template;
3. Store the template in a local repository, a central repository, or a portable token such as a smart card;
4. Live-scan the chosen biometric;
5. Process the biometric and extract the biometric template;
6. Match the scanned biometric against stored templates;
7. Provide a matching score;
8. Record a secure audit trail with respect to system use.

For example, you stand in front of a camera so it can capture your face or eye features. The system then extracts the appropriate features from the scan and stores the data as a template (feature vector). You then interact with the biometric device again, and the system verifies whether the data corresponds to the stored template or not. If the software fails to get a match, more tries may be needed. Once this procedure is complete, the system is operational (ie. training complete). When a

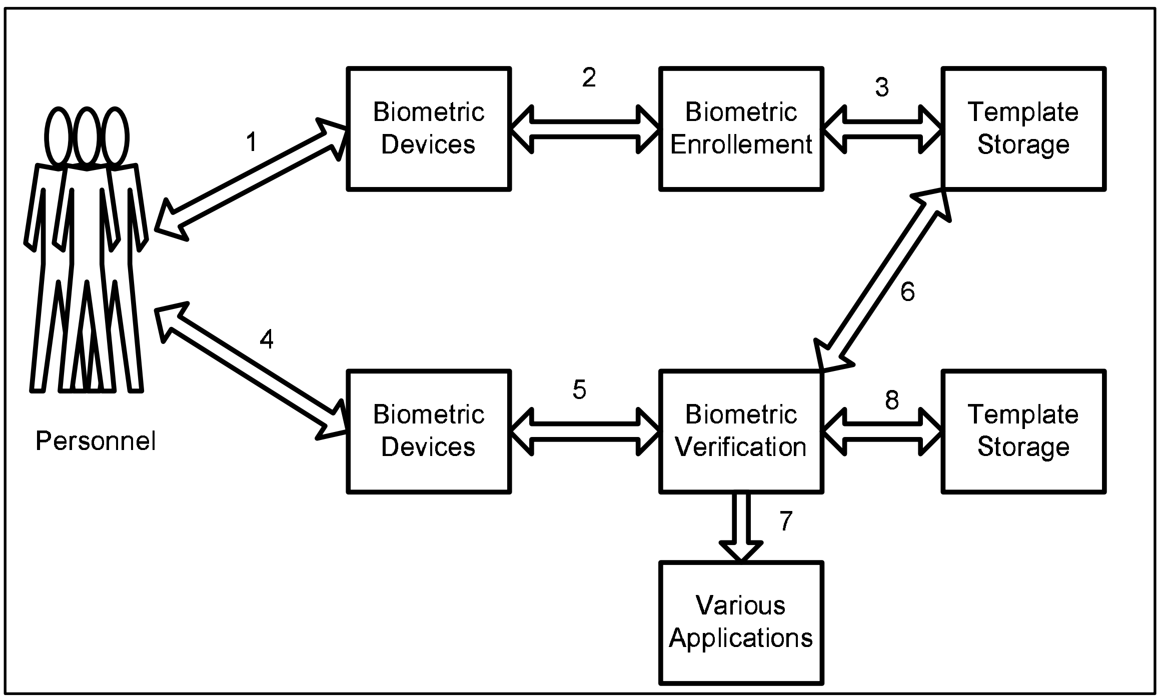

Figure 2.1: A typical biometric system

user tries to access the system, the user characteristics are scanned by whatever device is being used (user might be asked to supply a user-name as well), and the hardware passes the data to the software, which checks the user templates. If there is a match, the user is given access; otherwise, a message reports that the system can't identify the user. Biometric recognition scenarios can be classified into two types, (i) verification (or authentication) and (ii) identification (or recognition).

Biometric verification: is an one-to-one match that compares a query biometric image against a template biometric image whose identity is being claimed. To evaluate the verification performance, the verification rate (the rate at which legitimate users are granted access) vs. false acceptance rate (the rate at which impostors are granted access) is plotted. This curve is called the Receiver Operating Curve (ROC). A good verification system should balance these two rates based on appropriate operational needs.

Biometric identification: is a one-to-many matching process that compares a query biometric image against all the template images in the database to determine the identity of the query biometric. The identification of the test image is done by locating the image in the database that has the highest similarity with the test image. The identification method is a closed test, which means the sensor takes an observation of an individual that is known to be in the database. The test subjects (normalized) features are compared to the other features in the systems database and a similarity score is found for each comparison. These similarity scores are then numerically ranked in a descending order. The percentage of times that the highest similarity score corresponds to the correct match for all individuals is referred to as the top match score.

2.2 A Review of Various Biometrics

A number of biometric characteristics exist and are in use in various environments. Each biometric has its strengths and weaknesses, and the choice depends on the application. No single biometric is expected to effectively meet the requirements for all the applications. In other words, no biometric is optimal. The match between a specific biometric and an application is determined depending upon the operational mode of the application and the properties of the biometric characteristic. A brief introduction of the commonly used biometrics is given below:

1. *Fingerprint Recognition:* Fingerprint recognition systems use the distinctive patterns of the fingerprint to identify or verify identity (see Figure 2.2). The main features used for identification are the ridge endings, bifurcations, dots, crossovers, bridges, and their relative positions. There is a variety of approaches used in fingerprint verification. Some emulate the traditional police method of matching minutiae; others use straight pattern-matching algorithms. The most commonly used matching method is the Minutiae Based Matching. Two types of matching are possible: point matching and structure matching [2]. Neural Networks have also been proposed for classification. The input to the network is usually the low-pass filtered and averaged central region of the fingerprint. Given a certain fingerprint test pattern, the network assigns a match probability to each of the classes. With the advances made in

computing technology accompanied with lower prices, fingerprints recognition systems are gaining broader acceptance despite the common-criminal stigma. Besides the decreasing cost of fingerprint recognition systems, the main advantages are: ease of use, high levels of accuracy achieved, the wide range of deployments environments. Some of the disadvantages, however, are: 1. High false rejection rates, 2. Requirement of high-quality images, which is not always possible, 3. Injuries to fingers may affect successful recognition, 4. Need for complex feature extraction algorithms, 5. Inconsistent contact distortions due to mapping of 3D shapes onto 2D surfaces, and 6. Non-uniform contact factors like dryness, dirt, sweat, disease etc. result in noisy unreliable images.

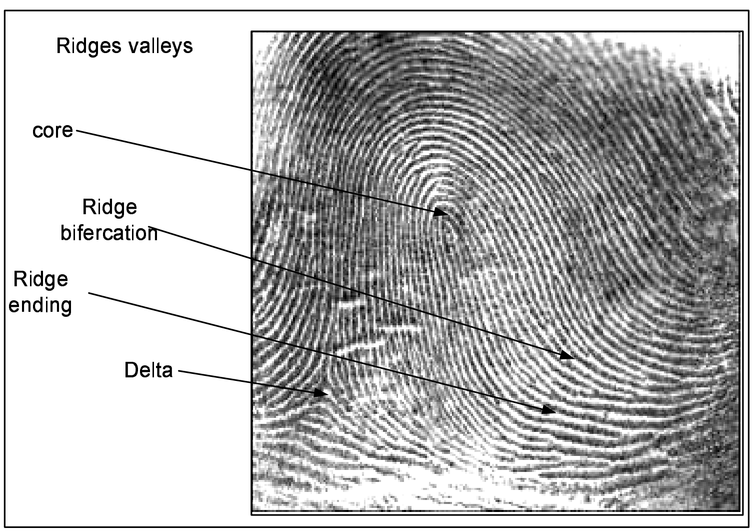

Figure 2.2: Details of a fingerprint [2]

2. *Face Recognition:* Face recognition is based on the analysis of facial characteristics. It requires a digital camera to capture a facial image of the user for

authentication. Such image is then stored as a template of the person. When authentication is required, the new image from the camera is processed and the new template is compared with the stored one. The matching can be done in either the space domain or some transform domain; features such as position of the eyes, nose, mouth, etc., have been used for this purpose. This technique has attracted considerable interest, although many people don't completely understand its full potential. Some vendors have made extravagant claims which are very difficult, if not impossible, to substantiate in practice. Because facial scanning needs an extra peripheral not customarily included with basic PCs, it is more of a niche market for network authentication. However, the casino industry has capitalized on this technology to create a facial database of scam artists for quick detection by security personnel. Current face recognition technology has a number of limitations including [2]: 1. The difficulty in recognizing faces from drastically different views and under illumination conditions, 2. It is questionable whether the face itself, without any contextual information, is a sufficient basis for recognizing a person from a large number of identities with an extremely high level of confidence, and 3. The face of a person changes with time. The main advantages of face recognition are as follows: 1. Cheap hardware, 2. Non intrusive, and very easy to use, and 3. Well developed techniques are available.

3. *Iris Recognition:* The iris is the plainly visible, colored ring that surrounds the pupil. It is a muscular structure that controls the amount of light entering the eye, with intricate details that can be measured, such as striations, pits, and furrows (see Figure 2.3). No two irises are alike. There is no correlation between the iris patterns of even identical twins, or the right and left eyes of the same individual. The amount of information that can be measured from a single iris is much greater than that from a fingerprint [3]. Unlike other biometric technologies that can be used in surveillance mode, iris recognition is an opt-in technology. In order to use the technology, the user must cooperate with the system.

   In iris recognition systems, the picture of the eye is first preprocessed to localize the inner and outer boundaries of the iris, and the eyelid contours, in order to extract just the iris portion. Eyelashes and reflections that may cover parts of the iris are detected and discounted. Advanced algorithms are then used to encode the iris pattern (such as the demodulation algorithm by Daugman) [16]. This creates a phase code for the iris texture, similar to a DNA sequence code. The demodulation process uses a 2-D wavelet function to represent the pattern with a 512 bytes code. In addition to being non-intrusive, iris recognition systems were shown to be accurate, and scalabile. The disadvantage is that recognition accuracy is affected by eye infections and wearing glasses. Accuracy is also affected by how good the collected images are. Note that public acceptance to this technology is still low.

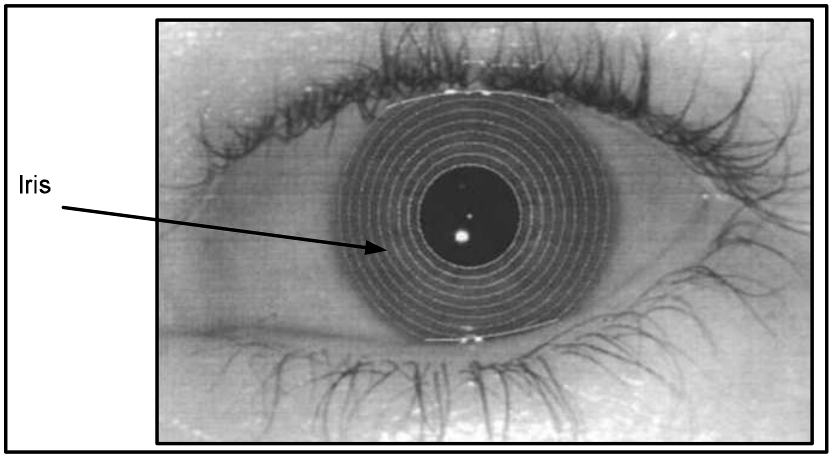

Figure 2.3: Location of the iris [3]

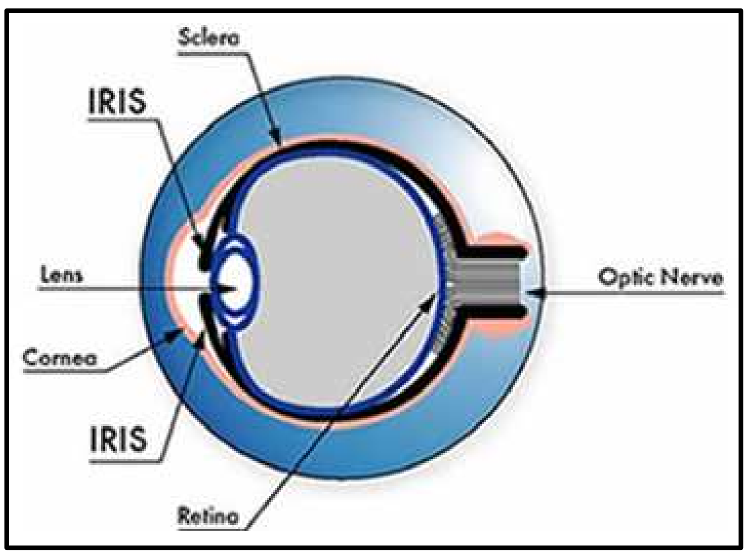

Figure 2.4: Location of the iris and the retina [3]

4. *Retina Recognition:* A retina-based biometric involves the analysis of the layer of the blood vessels situated at the back of the eye, as shown in Figure 2.5. It

involves the use of a low-intensity light source through an optical coupler to scan the unique patterns of the retina. Retinal scanning can be quite accurate but does require the user to look into a receptacle and focus on a given point. This is not particularly convenient if one is wearing glasses or reluctant to close contact with the reading device. For these reasons, retinal scanning is not warmly accepted by all users, even though the technology itself can work well. Advantages of retina biometrics include: 1. High accuracy, 2. Smaller storage requirements (35bytes of size). Some of the disadvantages are: 1. Expensive and intrusive (eye illuminated by bright light) 2. Recognition is susceptible to eye disease, 3. Several images are needed for registration.

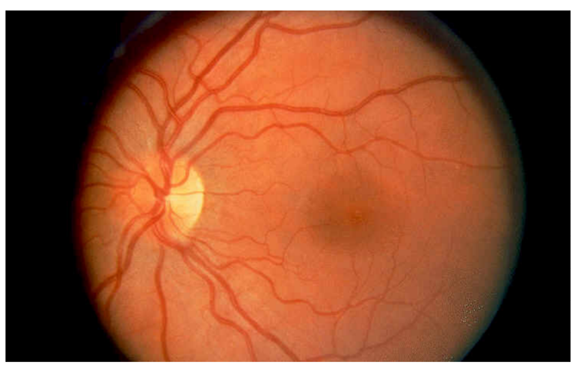

Figure 2.5: A sample image of a retinal scan

5. *Hand Geometry:* To use the system, the individual aligns his/her hand according to the guide marks on the hand reader hardware, and the reader captures a three-dimensional image of the fingers and knuckles and stores the data in a template, as shown in Figure 2.6. Hand geometry involves analyzing and

measuring the shape of the hand. Hand geometry offers good performance and is relatively easy to use. Hand recognition systems extract about 100 measurements of the length, width, thickness, and surface of the hand of four fingers. Hand recognition systems are usually used for verification where the measurements are compared to a template recorded during enrollment. Ease of integration into other systems and processes, coupled with ease of use, made hand geometry an obvious first step for many biometric projects. Hand geometry has been around for several years, and was adopted at the 1996 Olympic Games. Hand recognition systems are mainly used for verification and can achieve an equal error rate of 0.1%.

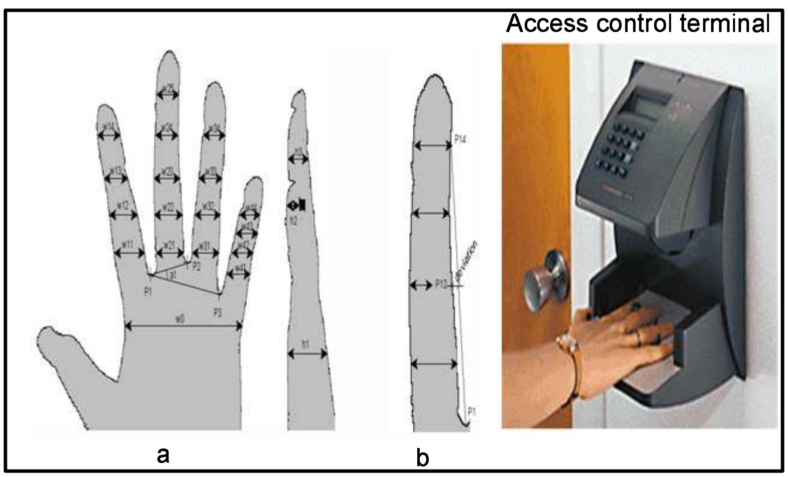

Figure 2.6: A typical recognition hand geometry system [4]

6. *Voice Recognition:* These systems are used in capturing the speaker's voice as well as his linguistic behaviors. Such systems are also used in voice-to-print authentication, where complex technology transforms voice into text. Voice

biometrics has the most potential for growth as it requires no new hardware since most PCs already have microphone input. However, poor quality and ambient noise can severely affect verification performance. In addition, the enrollment procedure has often been more complicated than with other biometrics, leading to the perception that voice verification is not user friendly! Its primary use continues to be telephone-based security applications. Its accuracy can be affected by a number of factors including noise and voice variations due to illness and fatigue. One obvious problem with voice recognition is fraud as the system can always be fooled using a tape of someone else's voice.

7. *Signature Recognition:* Such systems have a wide pubic acceptance. Signature recognition systems, also called dynamic signature verification systems, go far beyond simply looking at the shape of a signature (see Figure 2.7 a. shows a signature before processing. b. shows the signature after preprocessing, it has been smoothed and re-sampled). Signature verification analyzes the way a user signs his/her name, signing features such as speed, velocity, and pressure are as important as the finished signature's static shape. Signature verification enjoys a synergy with existing processes that other biometrics do not. People are used to signatures as a means for transaction related identity verification, and most would see nothing unusual in extending this to encompass biometrics. Signature verification devices are reasonably accurate

in operation and obviously lend themselves to applications where a signature is an accepted identifier. The problem is that our signatures vary significantly over time, so high accuracy requires multiple samples and an advanced analysis system. Surprisingly, relatively few significant signature applications have emerged compared to other biometric methodologies.

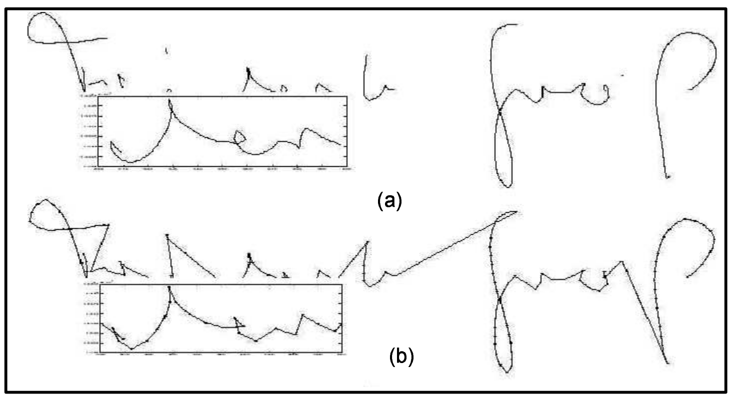

Figure 2.7: On-line processing of signatures [5]

8. *DNA:* Deoxyribo Nucleic Acid (DNA) has been proved to be the ultimate unique code to represent an individual, (see Figure 2.8). It is, however, currently used mostly in the context of forensic applications for person recognition. Three issues limit the usefulness of this biometrics for other applications: (i) contamination and sensitivity; (ii) automatic real-time recognition issues (the present technology for DNA matching requires cumbersome chemical methods (wet processes) involving an expert's skills and is not geared for on-line non-invasive recognition); (iii) privacy issues: information about susceptibilities of a person to certain diseases could be gained from the DNA

pattern and there is a concern that the unintended abuse of genetic code information may result in discrimination, e.g., in hiring practices.

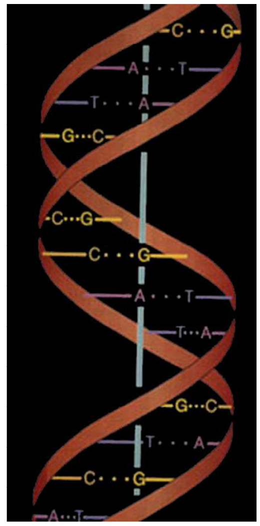

Figure 2.8: Pictorial representation of the DNA [6]

9. *Other Biometrics:* Apart from the above mentioned biometrics, there are few others that are either appropriate for a limited number of applications or are still under development. These include biometrics like Ears, Facial Thermograms, Hand Thermograms, Hand Veins, Gait, etc. The pattern of the heat radiated by the human body is a characteristic that can be captured by an infrared camera in an unobtrusive way much like a regular (visible spectrum) photograph. The technology could be used for covert recognition. A thermogram-based system does not require contact and is non-invasive, but image acquisition is challenging in uncontrolled environments, where heat emanating surfaces (e.g., room heaters and vehicle exhaust pipes) are present in

the vicinity of the body. Biometrics which uses infrared technology are Facial Thermogram, Hand Thermogram, Hand Vein, etc. Gait is the peculiar way one walks and is a complex spatio-temporal biometric. Gait is not supposed to be very distinctive, but is sufficiently discriminatory to allow verification in some low-security applications. Gait is a behavioral biometric which may not remain invariant, especially over a long period of time, due to fluctuations in body weight, major injuries involving joints or brain, or due to inebriety. Acquisition of gait is similar to acquiring a facial picture and, hence, may be seen as an acceptable biometric. Since gait-based systems use the video-sequence footage of a walking person to measure several different movements of each articulate joint; this system is input intensive and computationally expensive.

2.2.1 Selecting the Right Biometric Technology

Choosing the right biometric for a certain business application is a difficult task. A number of parameters need to be considered before choosing a certain biometric technology. While retinal scans, for example, are very accurate, the reluctance of common people in using such a technology makes it only useful in certain environments such as highly secured sensitive areas. Different biometric systems are appropriate for different applications, depending on perceived user profiles, the need to interface with other systems or databases, environmental conditions, and a host of other application-specific parameters.

However, any human physiological and/or behavioral characteristic can be used as a biometric characteristic as long as it satisfies the following requirements [12]:

a. *Universality*: each and every person throughout the world should have a biometric characteristic on the basis of which he/she could be recognized, for example every person has a thumb impression which is unique with him.

b. *Distinctiveness:* any two persons should be sufficiently different in terms of a biometric characteristic measure. Taking the above example, any two persons will have different thumb impressions.

c. *Collectability:* the biometric characteristic can be measured quantitatively with an ease. For example taking a snap of a face is typically easy with a camera, whereas it is not that easy to take a retina sample of a person.

d. *Permanence:* the characteristic should be sufficiently invariant (with respect to the matching criterion) over a period of time. Taking an example of face recognition as a person grows in age, his or her face changes over a time.

e. *Performance:* it refers to the achievable recognition accuracy and speed, the resources required to achieve the desired recognition accuracy and speed, as well as the operational and environmental factors that affect accuracy and speed.

f. *Acceptability:* it indicates the extent to which people are willing to accept the use of a particular biometric identifier (characteristic) in their daily lives.

g. *Circumvention:* it reflects how easily the system can be fooled using fraudulent methods or could be avoided. For example, in voice recognition, any person's voice can be recorded and the biometric system can easily be fooled.

Based on the above parameters a comparison table for various biometric technologies has been developed (Table 2.1) where High, Medium, and Low are denoted by H, M, and L, respectively.

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
DNA	H	H	H	L	H	L	L
Ear	M	M	H	M	M	H	M
Face	H	L	M	H	L	H	H
Facial thermogram	H	H	L	H	M	H	L
Fingerprint	M	H	H	M	H	M	M
Gait	M	L	L	H	L	H	M
Hand geometry	M	M	M	H	M	M	M
Hand vein	M	M	M	M	M	M	L
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Odor	H	H	H	L	L	M	L
Palmprint	M	H	H	M	H	M	M
Retina	H	H	M	L	H	L	L
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

Table 2.1: Comparison of various biometric techniques [12].

2.3 Multimodal Biometric Systems

Some of the limitations imposed by unimodal biometric systems can be overcome using multiple biometric modalities (such as face and fingerprint of a person or multiple fingers of a person). Such systems, known as multimodal biometric systems [17], are expected to be more reliable due to the presence of multiple, independent pieces of evidence [18]. These systems are also able to meet the stringent performance requirements imposed by various applications [7]. Multimodal biometric systems address the problem of non-universality, since multiple traits ensure sufficient population coverage. Further, multimodal biometric systems provide anti-spoofing measures by making it difficult for an intruder to simultaneously spoof the multiple biometric traits of a legitimate user. By asking the user to present a random subset of biometric traits (e.g., right index and right middle fingers, in that order), the system ensures that a live user is indeed present at the point of data acquisition. Thus, a challenge-response type of authentication can be facilitated using multimodal biometric systems. Combinations of face and fingerprint recognition have been proved to give better performance than individual biometrics as shown in Figure 2.9. Another scheme which used palm geometry and face recognition was also reported to yield good results [19]. However, an effective fusion scheme is necessary to combine the information presented by multiple domain experts.

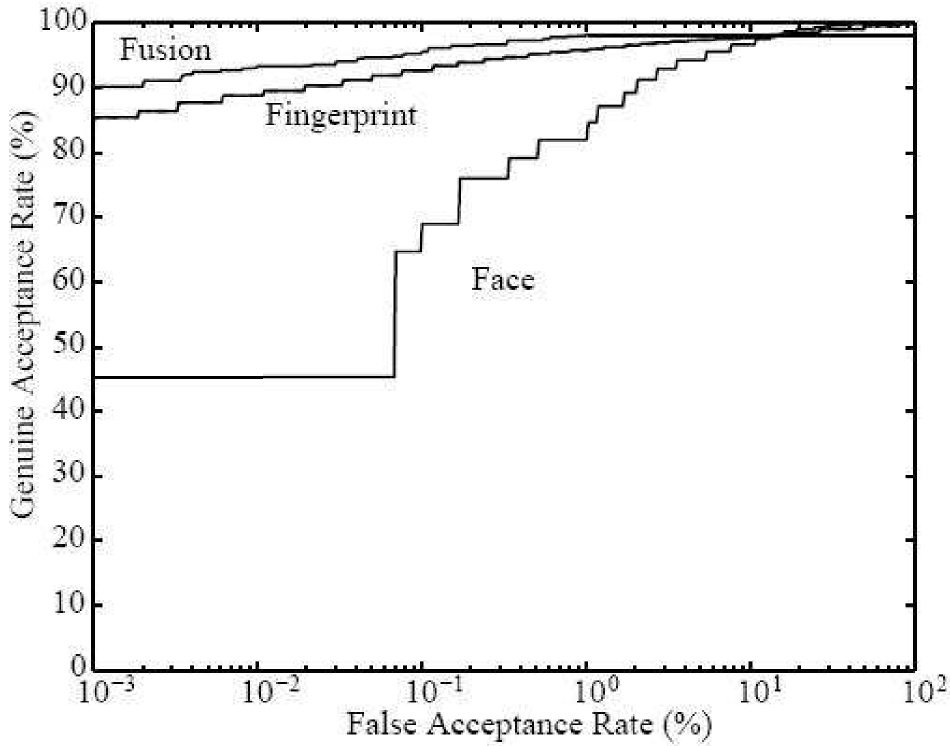

Figure 2.9: Matching accuracy using a combination of face and fingerprint[7].

2.4 Face Recognition Techniques

Face recognition has received a considerable attention in recent years both from the industry and research communities. Among the popular biometric technologies, facial features and face recognition scored the highest compatibility in a machine readable travel documents (MRTD) [8] system based on a number of evaluation factors (see Figure 2.10).

But automatic face recognition systems need to overcome various hurdles like pose invariance, illumination invariance, orientation invariance etc,. One problem with face recognition is that it is not an exceptionally accurate technique, it leaves

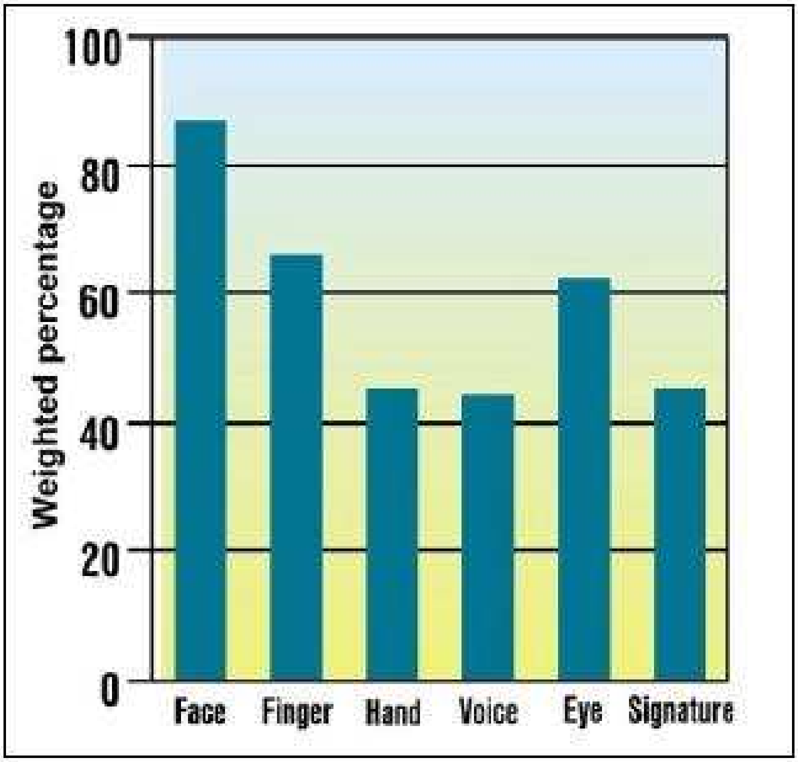

Figure 2.10: Comparison of various biometric features based on MRTD [8]

a small margin for error. Over the last decade there has been a tremendous effort put towards improving the performance of face recognition systems. As a result various novel techniques have been proposed ranging from the traditional template matching to the latest 3-dimensional techniques. It is only recently that face recognition became more popular in public areas such as airport surveillance, passport authentication, driving license, mobile phones, network login etc. The task of implementing face recognition in real life situations is becoming a reality. In this section we first identify the different approaches used in face recognition, followed by a brief description of each of the approaches.

2.4.1 Different Approaches to Face Recognition

The current face recognition techniques can be classified into four main categories based on the way these represent and identify the face: (i) *Appearance based or Holistics method* (using whole face region); (ii) *Model based methods* which employ shape and texture of the face, along with 3D depth information; (iii) *Template based face recognition;* and (iv) *Techniques using Neural Networks.* The list of available techniques is displayed Figure 2.11 [1].

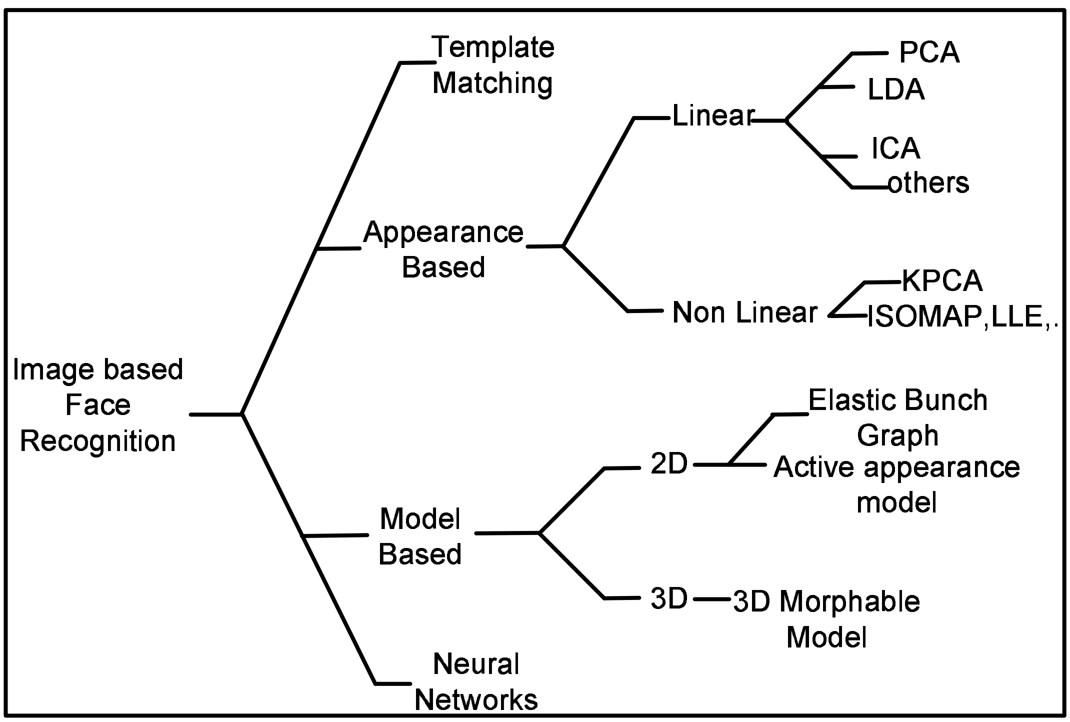

Figure 2.11: Classification of face recognition methods

[1]PCA: Principal Component Analysis, LDA: Linear Discriminant Analysis, ICA: Independent Component Analysis, KPCA: Kernel Principal Component Analysis, SVM: Support Vector Machines, GA: Genetic Algorithm

2.4.2 Template Based Face Recognition

Template matching has long been used as a technique in digital image processing for finding small parts of an image which match a certain template image [20]. Template matching involves the use of pixel intensity information, either as original gray-level or processed to highlight specific aspects of the data. The template can either be the entire face or regions corresponding to general feature locations, such as eyes or mouth. The normalized cross-correlation coefficient is used to identify the best match, defined as:

$$\rho_{I_T,T} = \frac{E(I_T T) - E(I_T)E(T)}{\sigma(I_T)\sigma(T)} \tag{2.1}$$

where I_T is the random variable with observations being the pixel values of the patch of image **I**. T is the random variable with observations being the pixel values of the template and σ represents the standard deviation and $E(.)$ is the expectation operator. Brunelli and Poggio [21] compared feature and template based methods directly with the same database of frontal face views. Their template matching strategy was based on earlier work, except that they automatically detected and used feature based templates of mouth, eyes and nose, in addition to the whole face. They showed that template based techniques can outperform appearance based approaches in a number of situations [21].

Rauschert et at. [22] proposed a face recognition method by means of template matching in frequency domain. A hardware based design is considered for special applications. Compared to the most common algorithms, template matching in the frequency domain was shown to be the most suitable approach [22].

Xiaoyan et al. [23] described a complete face recognition system using a template matching approach. They proposed a template matching approach along with a novel training algorithm for tuning the performance of the system to solve two types of problems simultaneously: 1) correct classification experiments which correctly recognize and identify individuals who are in the database; and 2) false positive experiments which reject individuals who are not part of the database. They showed that this type of training is capable of consistently producing high correct classification rates and low false positive rates.

2.4.3 Appearance Based Face Recognition

For this type of approach, the faces are stored as two dimensional intensity matrices. Each image is represented as a point in a high-dimension vector space. In order to identify different faces, an efficient representation (feature space) of the faces is derived. Given a test image, the similarity between the stored prototypes and the test view is then carried out in the feature space. Such approach is further classified into Linear (subspace) Analysis and Nonlinear (manifold) Analysis techniques.

Linear (Subspace) Analysis: Principal Component Analysis (PCA) [24, 25], Independent Component Analysis (ICA) [26, 27, 28, 29, 30, 31, 32], and Linear Discriminant Analysis (LDA) [33, 34] are classical linear subspace analysis techniques proposed for face recognition. Each technique has its own representation of high dimensional face vector space called basis vectors. For each case, the basis vectors are formed according to some statistical assumptions. After forming the basis vectors from the face database, the training faces are then projected onto the basis vectors to get the feature vectors. A distance measure is used to find the distance between the transformed test and the training images. The smaller the distance, the better the match is.

The above mentioned techniques can be considered as a linear transformation [8] of the original image space:

$$\mathbf{Y} = \mathbf{W}^T \mathbf{X} \tag{2.2}$$

where $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$ represent the $n \times N$ *data matrix* [2] where each $\mathbf{x}_i$ is the vector of dimension n, which is formed by concatenating the columns of a $p \times p$ face image matrix, where $n = p \times p$. Here n represents the total number of pixels in the face image, and N is the number of different face images in the training set. $\mathbf{Y}$ is the $d \times N$ feature vector matrix, d is the dimension of the feature vector and $\mathbf{W}$ is the transformation matrix with dimensions $(n \times d)$. Note that $d << n$. PCA, LDA and ICA are described in details in the next chapter.

Non-linear Analysis: The non-linear analysis technique is more complicated than linear models. Actually, linear subspace analysis is an approximation of non-linear manifold.

[2] *data matrix:* is a matrix containing all training images arranged in columns

A Kernel Principal Component Analysis (KPCA) was proposed as a nonlinear extension of a PCA for face recognition by Kim et al [35]. The basic idea of their work is to first map the input space into a feature space via nonlinear mapping, then compute the principal components in that feature space. They adopt the Kernel PCA as a mechanism for extracting facial features.

In what follows, the Kernel Principal Component Analysis (KPCA) [36] is briefly introduced as an example of non-linear subspace analysis.

Kernel PCA The Kernel PCA is seen as a nonlinear mapping from the input space $\mathbf{R}^n$ to the feature space $\mathbf{R}^L$ denoted by $\mathbf{\Psi}(\mathbf{x})$ where L is larger than n. This mapping is defined implicitly by specifying the form of the dot product in the feature space.

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) \tag{2.3}$$

An example of a kernel commonly used is the Gaussian kernel:

$$\kappa(\mathbf{x}, \mathbf{y}) = \exp(-\frac{||\mathbf{x} - \mathbf{y}||}{2\sigma^2}) \tag{2.4}$$

Let us say we have a training set of faces represented as vectors $\mathbf{x}_i \in \mathbf{R}^n; i = 1 : N$. Then, we have the corresponding set of mapped data points in feature space $\phi(\mathbf{x_i})$. The centered points become:

$$\tilde{\phi}(\mathbf{x_i}) = \phi(\mathbf{x_i}) - \frac{1}{N}\sum_{i=1}^{N}\phi(\mathbf{x_i}) \qquad i = 1, 2, ..., N \tag{2.5}$$

The Kernel PCA algorithm is implemented as follows [36]:

1. Choose an appropriate Kernel function $\kappa(.,.)$.

2. Calculate the Kernel matrix from the mapped data:

$$K_{ij} \equiv \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) = \kappa(\mathbf{x}_i, \mathbf{x}_j) \qquad i, j = 1, 2, ..., N \tag{2.6}$$

3. Using $\mathbf{K}$, we construct the covariance matrix of the centered data:

$$\begin{aligned} \tilde{K}_{ij} &\equiv \tilde{\phi}(\mathbf{x}_i)\tilde{\phi}(\mathbf{x}_j) \qquad (2.7) \\ &= K_{ij} - \frac{1}{N}\sum_{p=1}^{N} K_{ip} - \frac{1}{N}\sum_{q=1}^{N} K_{qj} - \frac{1}{N^2}\sum_{p,q=1}^{N} K_{pp} \quad i, j = 1, 2, ..., N \end{aligned}$$

4. Find the set of eigenvectors $\{\mathbf{b}_\mathbf{i}^\alpha : i = 1, 2, ..., L\}$ of the matrix $\tilde{\mathbf{K}}$, which are our set of basis vectors $\mathbf{b}^\alpha$ in the feature space.

5. For a test point, $\mathbf{x}$ the unnormalized KPCA components, are then given by

$$\mathbf{P}^\alpha(\mathbf{x}) \propto \mathbf{b}^\alpha \cdot \phi(\mathbf{x}) = \sum_{\mathbf{i=1}}^{\mathbf{N}} \mathbf{b}_\mathbf{i}^\alpha \kappa(\mathbf{x}, \mathbf{x_i}) \tag{2.8}$$

The values of the components depend on the normalization taken for the set of vectors $\{\mathbf{b}^\alpha\}$, which are orthogonal but need not be orthonormal.

A comparison of different subspace algorithms was conducted (for d = 20) using a 5-fold cross-validation method [3], where d is the dimensionality of the subspace. The data was taken from the FERET database [37], and contained 1,829 images from 706 subjects. A summary of the results is given in Table 2.2 [8]. It was found that the KPCA method performs slightly better than both PCA and ICA algorithms. In terms of the computational complexity, PCA performs the best.

	PCA	ICA	KPCA
Accuracy	77%	77%	87%
Computation (floating point operation)	10^8	10^9	10^9
Uniqueness	Yes	No	Yes
Projection	Linear	Linear	Nonlinear

Table 2.2: Comparison of different subspace techniques [8]

2.4.4 Hybrid Appearance based techniques

The concept behind such techniques is to combine the results for a number of approaches to improve the overall accuracy. In what follows, we list few of such combinations.

Yang and co-workers [38] examined the theory of Kernel Fisher discriminant analysis (KFD) in a Hilbert space and developed a two-phase KFD framework, i.e., Kernel Principal Component Analysis (KPCA) plus Fisher linear discriminant analysis (LDA). This framework provided a novel insight into the nature of KFD. Based on such framework, the authors proposed a complete Kernel Fisher discriminant analysis (CKFD) algorithm. CKFD can be used to perform discriminant analysis in "double discriminant subspaces." The fact that, the technique uses the two kinds of discriminant information, regular and irregular, makes the CKFD a more powerful discriminator.

A face recognition approach based on KPCA and genetic algorithms (GAs) was also proposed by Zhang and Liu [39]. Using a polynomial function as a Kernel in KPCA, the nonlinear principal components can be obtained. After the nonlinear principal components are obtained, GAs are used to select the optimal feature set for classification. At the recognition stage, linear support vector machines (SVM) are used as classifiers.

Xiaoguang, Yunhong and Jain [40], used the sum rule and RBF-based integration strategies to combine three commonly used face classifiers based on PCA, ICA and LDA representations to achieve a robust face recognition system. Experiments conducted on a face database containing 206 subjects (2,060 face images) show that the proposed classifier combination approaches outperform individual classifiers.

2.4.5 Model Based Face Recognition

This approach uses a model of the face to perform recognition. The model is formed from prior knowledge of face features. A recent technique developed by Wiskott et al. [9] is the elastic bunch graph matching technique [6]. Also, Cootes et al., by integrating both shape and texture, developed a new technique called the 2D morphable face model which measures face variations [18]. A more advanced 3D morphable face model has also been explored to capture the true 3D structure of human face surface.

The model-based approach usually involves three steps: 1) Developing the face model, 2) Fitting the model to the given face image; 3) Using the parameters of the fitted model as the feature vector to calculate the similarity between the query face and the prototype faces from the database and perform the recognition.

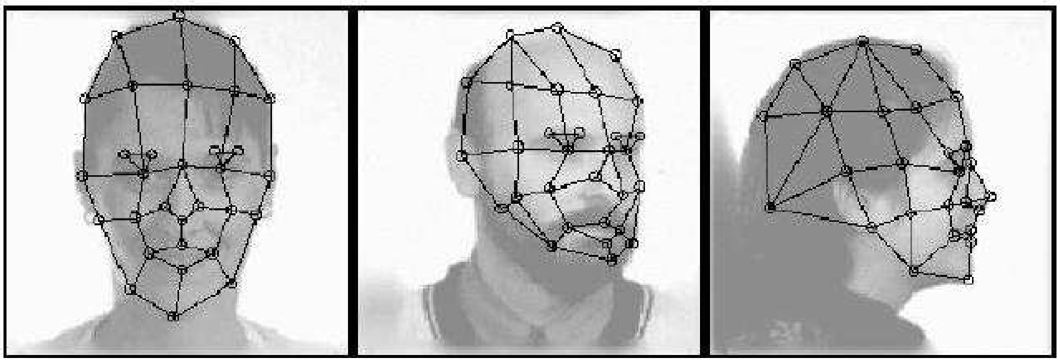

Figure 2.12: Multiview faces overlaid with labeled graphs [9]

Feature-based Elastic Bunch Graph Matching

All human faces share a similar topological structure. Wiskott et al. [9] presented a general class of recognition method for classifying members of a known class of objects. Faces are represented as graphs, with nodes positioned at fiducial points and edges labeled with 2-D distance vectors (see Figure 2.12). Each node contains a set of 40 complex Gabor wavelet coefficients, including both phase and magnitude, known as a jet. Face recognition is based on labeled graphs. A labeled graph is a set of nodes connected by edges; nodes are labeled with jets; edges are labeled with distances. Thus, the geometry of an object is encoded through the edges while the gray value distribution is patch-wise encoded through the nodes. Face Bunch Graph has a stack-like structure that combines graphs of individual sample faces.

This provides full combinatorial power of this representation even if it is constituted of only few graphs.

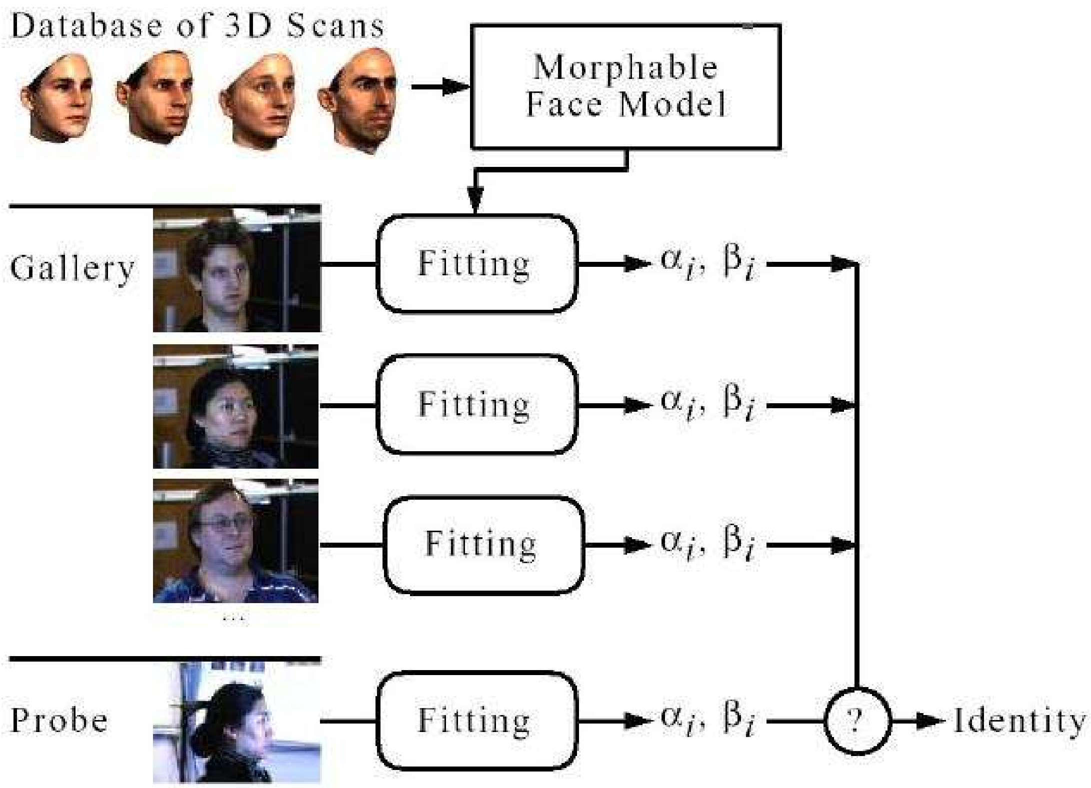

Figure 2.13: The 3-D morphable face models [10]

3D Morphable Model

This technique utilizes the fact that human face is a surface lying in the 3D space intrinsically. This implies in principle, that the 3D model is better for representing faces, especially for handling facial variations, such as pose, illumination. Blanz et al. [10] proposed a method based on a 3D morphable face model that encodes shape and texture in terms of a set of parameters, and an algorithm that recovers these parameters from a single image of a face(see Figure 2.13). Performance evaluation of Elastic Bunch Graph Matching were conducted on the FERET database. The results are given in Table 2.3. The recognition results of the system were good on identical poses, e.g., frontal views against frontal views. However, across different poses, e.g., frontal views against half profiles, the system performs rather poorly.

Basso et al. [41] introduced the new concept of regularized 3D morphable models, along with an iterative learning algorithm, by adding in the statistical model a noise/regularization term which is estimated from the training set. With regularized 3D morphable models, it is possible to handle missing information, as it often occurs with data obtained by 3D acquisition systems; additionally, the new models are less complex, but as powerful as the non-regularized ones.

2.4.6 Face Recognition using Neural Networks

The ability of neural networks (NN) to learn from their experience is the key element in the problem solving strategy of a pattern recognition task. A neural networks

			Testview		
			frontal(%)	side(%)	profile(%)
training view	frontal	mean	94	85	65
		std	6.3	20.7	18.2
	side	mean	89	90	70
		std	6.4	9.2	18.9
	profile	mean	71	71	84
		std	9.2	12.2	16.4

Table 2.3: Recognition results of 3D morphable models [13]

system can be seen as an information processing system composed of a large number of interconnected processing elements. Each processing element (also called node, neuron) calculates its activity locally on the basis of the activities of the cells to which it is connected. The strengths of its connections are changed according to some transfer function that explicitly determines the cell's output, given its input. The learning algorithm determines the performance of the neural networks system.

In [42], a new approach for face recognition using Eigenfaces and Neural Networks was proposed. An Eigenfaces technique is applied to extract the feature vectors which contain concise information about the face images. Using such feature vectors, the Neural networks is trained. Once it is trained, it is ready to recognize face images. Eight subjects (persons) were used in a database of 80 face images. A recognition accuracy of 95.6% was achieved with vertically oriented frontal views of human face.

Hazem [43] also proposed to use a Neural Networks technique to train the gray level images taken in real life situations. It was shown that as the number of individuals in the training database is increased from 5 to 9 the system's performance falls from 94.54% to 90.52%.

The major limitation of Neural Networks is that large number of training samples are required to train the system before it could give good accuracy and as the number of individuals increases the number of samples also needs to be increased.

2.5 Standard Databases

A number of face databases have been collected for different face recognition applications. Table 2.4 lists a selection of those available in the public domain. The last column of the table indicates the conditions in which the images were captured (** p- pose, e- expression, i- illumination, i/o- indoor/outdoor, t-time interval and o- occlusion). The ORL database also known as AT&T database contains 40

Face database	No. of subjects	No. of images	Variations include **
ORL	40	400	p,e
Yale	15	165	i,e
AR	> 120	$> 3,000$	i,e,o,t
MIT	16	432	i,p,s
UMIST	20	564	p
CMU PIE	68	41,368	p,i,e
XM2VTS	295	*	*
FERET	$> 30,000$	$> 120,000$	p,i,e,i/o,t

Table 2.4: List of available databases in the public domain [8]

subjects. Each subject in the database has ten images in varying pose and expressions. The CMU PIE database is collected with well constrained pose, illumination and expression. FERET and XM2VTS databases are the two most comprehensive databases, which can be used as benchmarks for detailed testing. The XM2VTS is especially designed for multi-modal biometrics, including audio and video cues. (*For each subject, the database collects video+audio+3D model. For details, see http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/). The FERET database was used for vendor test twice in the year 2000 and 2002 (namely FRVT2000 and

FRVT2002, the database was extended between 2000 and 2002). The FERET has a complicated protocol [40], which can be applied to different scenarios.

2.6 Chapter Summary

In this chapter, we described how typical biometric systems work and discussed the different requirements for such systems. We also discussed a number of biometrics including Fingerprint, Face recognition, Retina recognition, Hand geometry, Voice recognition, and DNA matching. We discussed the concept of multi-model biometrics. Next, we focused on face recognition and presented the various face recognition approaches available in the literature. Lastly, we listed a number of publicly available face databases.

Chapter 3

Linear Subspace Techniques

In data mining, we often encounter situations where we have a large number of variables in the database that need to be analyzed. In such situations it is likely that subsets of variables are highly correlated with each other. One of the key steps in data mining is finding ways to reduce dimensionality without sacrificing accuracy.

In feature extraction, all available variables are used and the data is transformed to a reduced dimension space. Thus, the aim is to replace the original variables by a smaller set of underlying variables. There are several reasons for performing feature extraction [2].

1. To reduce the bandwidth of the input data(with the resulting improvements in speed and reductions in data requirements);

2. To reduce redundancy;

3. To recover new meaningful underlying variables or features that describe the data, leading to greater understanding of the data generation process;

4. To produce a low-dimensional representation (ideally in two dimensions) with minimum loss of information so that the data may easily be viewed and relationships are structured in the data identified.

Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Independent Component Analysis (ICA) fall under the broad class of linear transformations that transform a number of (possibly) correlated variables into a (smaller) number of variables. The objective is to reduce the dimensionality (number of variables) of the dataset but retain most of the original variability in the data. As such, Linear Subspace Techniques are quite often seen as feature extraction techniques used to reduce or remove redundant or irrelevant information from the data.

The problem of dimensionality in face recognition arises when it is not known which measurement may be important for the application. That is, when there is inadequate knowledge about the structure of the data that may arise in an application. Our main attack to this problem is to study and improve classification methods, since the objective in classification is to provide a description that is well matched to the structure of the data.

In this work, the three algorithms namely the Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Independent Component Analysis (ICA) are implemented. PCA more famously known as “Eigenfaces for face recognition” [24] is discussed section 3.2, LDA and ICA techniques are discussed in section 3.3 and 3.4, respectively. We have used the leave one out evaluation technique of PCA, LDA and ICA on the Yale faces database [11]. Few sample faces of subjects are shown in Figure 3.1, Figure 3.2,and Figure 3.3. The Yale faces database contains images of 15 subjects in 11 different conditions, like smiling, wearing goggles, different brightness etc.

3.1 Vector Representation of Images

Image data can be represented as vectors, that is, as points in a high dimensional vector space. For example, an $n_1 \times n_2$ image $P(i, j)$ can be mapped to a vector $\mathbf{x} \in \mathbf{R}^n$, by lexicographic ordering of the pixel elements (such as by concatenating each row or column of the image). Once we have converted all images into columns, these columns can be arranged to form a matrix $\mathbf{X}$ called the *Data matrix*.

Notice that if the image is of size 100×100, the resulting vector is of size $10,000 \times 1$; a very high dimensional vector. Despite this high dimensional embedding, the natural constraints of the physical world (and the imaging process) dictate that the data will, in fact, lie in a lower-dimensional (though possibly disjoint) manifold. The primary goal of subspace analysis is to identify, represent, and parameterize such manifold in accordance with some optimality criteria.

Figure 3.1: Samples of different subjects [11]

Figure 3.2: Samples of one subject in different conditions [11]

Two important statistical measures often used in this thesis are the Mean and the Covariance. These are defined below:

Mean: We can treat the columns of data matrix $\mathbf{X}$ as random samples associated to random vector $\mathbf{x}$. Then mean vector of the random vector is defined as

$$\mathbf{m} = E[\mathbf{x}] \tag{3.1}$$

where $E[.]$ is the expected value of the argument. An estimate of the equation 3.1 can be found from the samples of random vector $\mathbf{x}$, which are the columns of data

Figure 3.3: Samples of test faces used in evaluation of the PCA and the LDA [11]

matrix, $\mathbf{x}_i$, using the expression.

$$\mathbf{m} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \tag{3.2}$$

The mean vector evaluated from the data matrix represents a mean face for the database when converted from column to image matrix. The mean face for the Yale faces database is shown in Figure 3.4.

Figure 3.4: Mean face from the training database

Covariance Matrix: The covariance matrix of the random vector $\mathbf{x}$ is defined as:

$$\mathbf{C} = E[(\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T] \tag{3.3}$$

Given the data matrix containing the samples associated with random vector $\mathbf{x}$, the covariance matrix can be estimated using:

$$\mathbf{C} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T \tag{3.4}$$

3.2 Principal Component Analysis

Principal Component Analysis (PCA) has been successfully used in a wide range of applications, including data mining, financial data analysis, image compression, and face recognition, to mention a few. Principal Component Analysis is defined as a dimensionality reduction technique which transforms a random vector say $\mathbf{x}$, say of size n, to a random vector of $\mathbf{y}$, say of size k. Where k is chosen smaller than n. This transformation is defined below:

Let $\mathbf{e}_i$ and λ_i, $i = 1, 2, ..., n$ be the eigenvectors and corresponding eigenvalues of the covariance matrix $\mathbf{C}_x$ of the random vector $\mathbf{x}$ (where $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n$). Since we know that $\mathbf{x}$ takes real values (eg. image data). The covariance matrix $\mathbf{C}_x$ is real and symmetric. It follows that the eigenvalues of $\mathbf{C}_x$ are real. A transformation matrix is formed whose columns are the eigenvectors of $\mathbf{C}_x$ which is given by:

$$\mathbf{W} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_n] \tag{3.5}$$

Principal Component Analysis is defined by a transformation obtained as follows:

$$\mathbf{y} = \mathbf{W}^T(\mathbf{x} - \mathbf{m}_x) \tag{3.6}$$

The transformation given by equation 3.6 [1] has several important properties. The first property we examine here is the covariance matrix of the random vector $\mathbf{y}$. This is defined as [44]:

$$\mathbf{C}_y = E[(\mathbf{y} - \mathbf{m}_y)(\mathbf{y} - \mathbf{m}_y)^T] \tag{3.7}$$

where $\mathbf{m}_y$ is equal to zero vector $\mathbf{0}$, since:

$$\begin{aligned} \mathbf{m}_y &= E[\mathbf{y}] \\ &= E[\mathbf{W}^T(\mathbf{x} - \mathbf{m}_x)] \\ &= \mathbf{W}^T E[\mathbf{x}] - \mathbf{W}^T \mathbf{m}_x \\ &= \mathbf{0} \end{aligned} \tag{3.8}$$

By substituting 3.6 and 3.8 into 3.7 gives the following expression for $\mathbf{C}_y$ in terms of $\mathbf{C}_x$:

$$\begin{aligned} \mathbf{C}_y &= E[(\mathbf{W}^T\mathbf{x} - \mathbf{W}^T\mathbf{m}_x)(\mathbf{W}^T\mathbf{x} - \mathbf{W}^T\mathbf{m}_x)^T] \\ &= E[\mathbf{W}^T(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T\mathbf{W}] \\ &= \mathbf{W}^T E[(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T]\mathbf{W} \\ &= \mathbf{W}^T\mathbf{C}_x\mathbf{W} \end{aligned} \tag{3.9}$$

where last step is from the definition of the covariance matrix.

It is shown by Lawley and Maxwell [45] that $\mathbf{C}_y$ is a diagonal matrix with

[1] This transformation is also know as Hotelling transform.

elements equal to the eigenvalues of $\mathbf{C}_x$; that is

$$\mathbf{C}_y = \begin{bmatrix} \lambda_1 & & & & 0 \\ & \lambda_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ 0 & & & & \lambda_n \end{bmatrix}$$

This is an important property, since the terms other than the main diagonal are 0, the elements of $\mathbf{y}$ are uncorrelated. In addition, each eigenvalue λ_i is equal to the variance of the i^{th} element of $\mathbf{y}$.

The second important property deals with reconstruction of random vector $\mathbf{x}$ from random vector $\mathbf{y}$. Since we consider $\mathbf{x}$ whose observations are real, the covariance matrix $\mathbf{C}_x$ is real. It follows that the set of eigenvectors of $\mathbf{C}_x$ form an orthonormal basis, $\mathbf{W}^{-1} = \mathbf{W}^T$. Using this property, $\mathbf{x}$ can be reconstructed from $\mathbf{y}$ by using the relation:

$$\mathbf{x} = \mathbf{W}\mathbf{y} + \mathbf{m}_x \tag{3.10}$$

Suppose, however, that instead of using all eigenvectors of $\mathbf{C}_x$, we construct $\mathbf{W}$ from the first k eigenvectors corresponding to the largest eigenvalues. The $\mathbf{y}$ vector will then be k dimensional and the reconstruction giving by equation 3.10 is no longer exact. It is given as follows:

$$\hat{\mathbf{x}} = \mathbf{W}_k\mathbf{y} + \mathbf{m}_x \tag{3.11}$$

$\hat{\mathbf{x}}$ represents an approximation of $\mathbf{x}$ obtained from the transformation matrix $\mathbf{W}$ composed of first k eigenvectors of $\mathbf{C}_x$.

The mean square error between $\mathbf{x}$ and $\hat{\mathbf{x}}$ is given by the expression [44]:

$$\begin{aligned} e_{ms} &= \sum_{j=1}^{n} \lambda_j - \sum_{j=1}^{k} \lambda_j \\ &= \sum_{j=k+1}^{n} \lambda_j \end{aligned} \tag{3.12}$$

The first line of equation 3.12 indicated that the error is zero, if $k = n$. Additionally since the λ_j's decrease monotonically, equation 3.12 also shows that the error can be minimized by selecting the k eigenvectors associated with the largest eigenvalues. Thus PCA is optimal in the sense that it minimizes the mean square error between the vector $\mathbf{x}$ and its approximation $\hat{\mathbf{x}}$.

The Eigenfaces algorithm in [24] is a decomposition algorithm based on Principal Component Analysis. The Eigenfaces algorithm finds the feature vectors which best accounts for the distribution of face images within the entire face database.

Recognition of images using PCA takes three basic steps. The transformation matrix is first created using the training images. Next, the training images are projected onto the matrix columns. Finally, the test images are identified by projecting these into the subspace and comparing them to the trained images in the subspace domain. The PCA algorithm is outlined as follows:

1. Creating Matrix $\mathbf{W}$: The following steps are carried to compute matrix $\mathbf{W}$

   - **Removing the Mean:** Each of the training images represented as columns of the data matrix $\mathbf{X}$ (refer to section 3.1) is first mean adjusted to obtain the following matrix:

$$\mathbf{P} = (\mathbf{x}_1 - \mathbf{m}, \mathbf{x}_2 - \mathbf{m}, ..., \mathbf{x}_K - \mathbf{m}) \tag{3.13}$$

where $\mathbf{m}$ is now an estimate of the mean:

$$\mathbf{m} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \tag{3.14}$$

It is to be noted that the mean vector, when converted back to a matrix of size identical to that of the original images, represents a face image called *mean face*. The mean face for the Yale database [11] is shown in Figure 3.4.

- **Estimate the Covariance Matrix:** An estimate of the covariance matrix can then be obtained when matrix $\mathbf{P}$ multiplied by its transpose:

$$\mathbf{Q} = \frac{1}{N}\mathbf{P} \times \mathbf{P}^T \tag{3.15}$$

- **Computing the eigenvalues and eigenvectors of Q:** The matrix $\mathbf{Q}$ is of size size $n \times n$ (example 10000×10000). It is difficult if not impossible to compute the eigenvectors for such a large matrix. This problem is fortunately solved using the method presented by Turk and Pentland [12].

  We first find the reduced covariance matrix $\tilde{\mathbf{Q}}$ (instead of $\mathbf{Q}$, refer to equation 3.15):

$$\tilde{\mathbf{Q}} = \frac{1}{N}\mathbf{P}^T \times \mathbf{P} \tag{3.16}$$

where $\tilde{\mathbf{Q}}$ has dimension of $N \times N$ where N is the number of faces in the training set. The N non-zero eigenvalues (sorted in descending order) and the corresponding eigenvectors of $\mathbf{Q}$ can be derived from the N eigenvalues and eigenvectors of $\tilde{\mathbf{Q}}$ as:

$$\begin{aligned}
\tilde{\mathbf{Q}}\tilde{\mathbf{e}}_i &= \tilde{\lambda}_i\tilde{\mathbf{e}}_i \qquad i = 1, 2, ..., N \\
\tfrac{1}{N}\mathbf{P}^T\mathbf{P}\tilde{\mathbf{e}}_i &= \tilde{\lambda}_i\tilde{\mathbf{e}}_i \\
\tfrac{1}{N}\mathbf{P}\mathbf{P}^T\mathbf{P}\tilde{\mathbf{e}}_i &= \tilde{\lambda}_i\mathbf{P}\tilde{\mathbf{e}}_i \\
\mathbf{Q}\mathbf{P}\tilde{\mathbf{e}}_i &= \tilde{\lambda}_i\mathbf{P}\tilde{\mathbf{e}}_i
\end{aligned} \tag{3.17}$$

The eigenvalue decomposition of $\mathbf{Q}$:

$$\mathbf{Q}\mathbf{e}_i = \lambda_i\mathbf{e}_i \qquad i = 1, 2, ..., N \tag{3.18}$$

comparing 3.18 and 3.17 we have:

$$\lambda_{\mathbf{i}} = \tilde{\lambda_{\mathbf{i}}} \qquad i = 1, 2, ..., N \tag{3.19}$$

$$\mathbf{e_i} = \mathbf{P}\tilde{\mathbf{e_i}} \qquad i = 1, 2, ..., N \tag{3.20}$$

- **Selecting the largest** K **eigenvectors:** Each $\mathbf{e_i}$ from the above represents an eigenface, the first six eigenfaces are shown in Figure 3.5. Keep only the eigenvectors associated with the largest K eigenvalues (selection of K is discussed latter). Form a matrix containing these eigenvectors as:

$$\mathbf{W} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_K] \tag{3.21}$$

Figure 3.5: The first six eigenfaces

2. **Projecting the training images:** All mean adjusted images from the training database are projected into the eigenspace. To project an image into the eigenspace, calculate the dot product of the image with each of the selected eigenvectors, that is,

$$\mathbf{Y} = \mathbf{W}^T\mathbf{P} \tag{3.22}$$

Note: $\mathbf{P}$ is the $n \times N$ matrix. $\mathbf{Y}$ is $K \times N$ feature matrix, where K is the dimension of feature vectors, and $\mathbf{W}$ is the $n \times K$ transformation matrix. Note that $K << n$ for example $K = 20$ and $n = 10,000$. *Hence the resulting feature in vectors in* $\mathbf{Y}$ *are very small in dimension compared to the data vectors in* $\mathbf{X}$.

3. **Identifying test images:** Each test image is first mean adjusted by subtracting the mean image, and then projected into the same eigenspace defined above. The projection of the test image is compared to the projections of the training images. A distance measure is then used to match the test image to

the training image in the feature space giving the least distance. A number of similarity measures have been used in the literature; the most common one is the Euclidean distance [24]. (see chapter 4 for more details.)

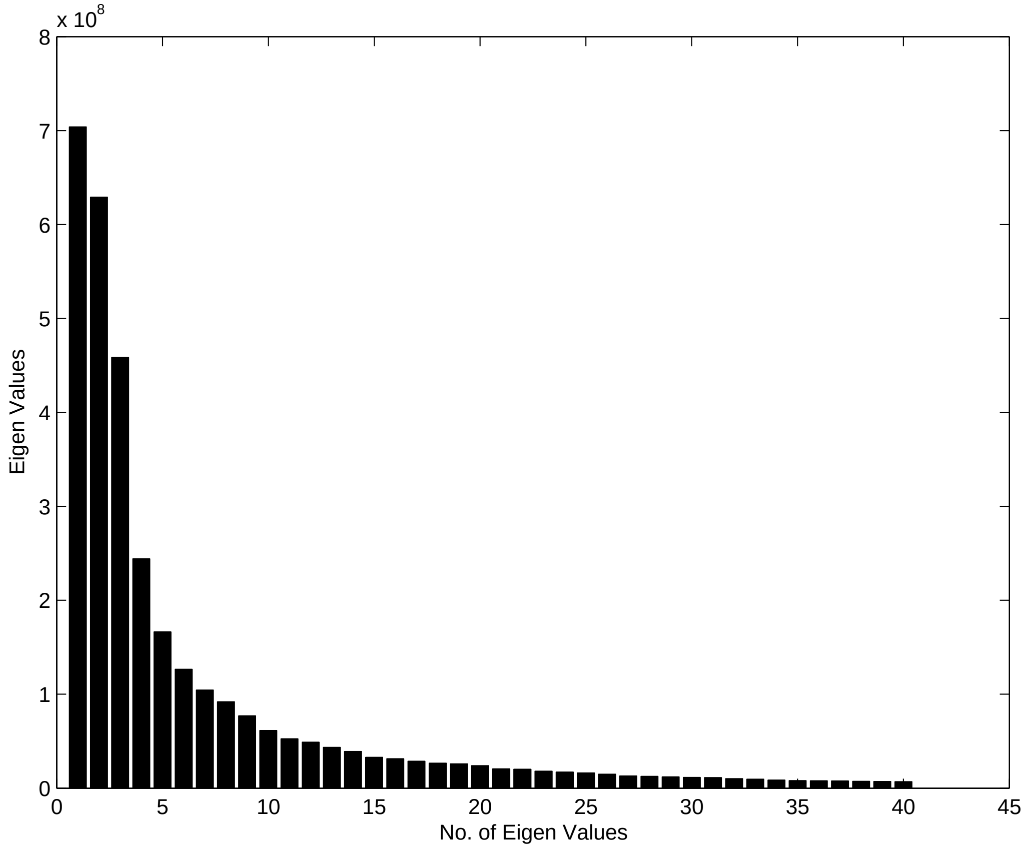

Figure 3.6: The eigenvalue spectrum

3.2.1 Experimental Results

We have performed a Leave-one-out experiment on the Yale faces database [11]. The Yale faces database contains images of 15 subjects in 11 different conditions. These 11 conditions are centerlight, glasses, happy, leftlight, noglasses, normal, rightlight, sad, sleepy, surprise, wink. *The Leave-one-out experiment is based on choosing the test images belonging to a condition (say centerlight) from each person, then train the leftover 10 images in the database belonging to each person using the PCA algorithm and perform face recognition. This process is repeated for the 11 different condition, by choosing a different condition each time. The final recognition accuracy is evaluated by finding the average recognition accuracy for all conditions.*

The plot of eigenvalues shown in Figure 3.6 is called the eigenvalues spectrum. The eigenvalue represent the variance of data when projected onto the corresponding eigenvector. We infer from the Figure 3.6 that, the first few (eg. 15-20) eigenvectors are most important, since they represent most of the data variation. The columns of the transformation matrix $\mathbf{W}$ represent eigenfaces, shown in Figure 3.5. We notice that 20 eigenfaces (maximum value for K) are enough to reach the maximum recognition accuracy of 83.03% (refer to Figure 3.7).

3.3 Linear Discriminant Analysis

Linear Discriminant Analysis has been successfully used as a classification technique for a number of problems, including speech recognition; face recognition, and multimedia information retrieval. While PCA takes the complete training data as one entity, LDA's goal is to find an efficient way to represent the face vector space by

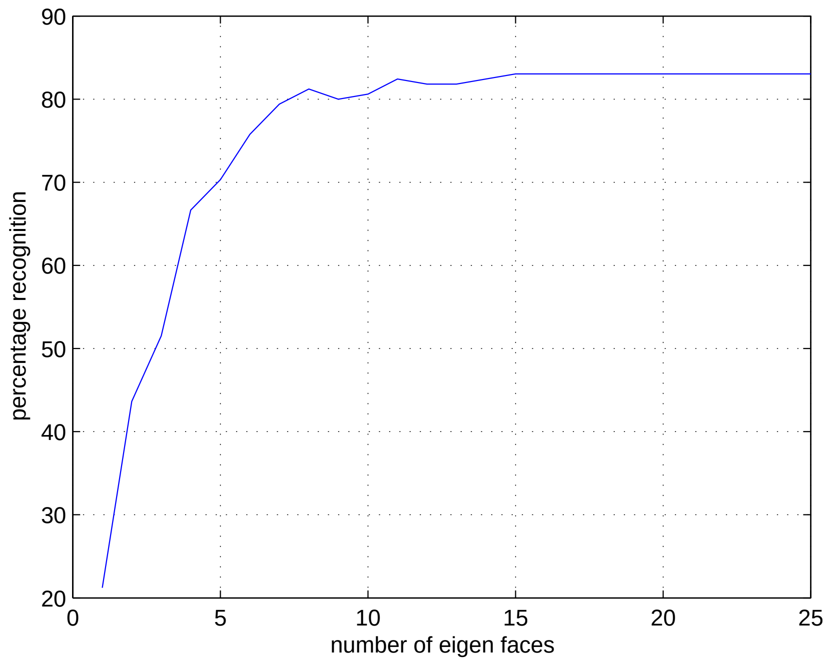

Figure 3.7: Recognition accuracy using PCA

exploiting class [2] information.

Using Linear Discriminate Analysis, we desire to find a linear transformation from the original image space to the reduced dimension feature space as:

$$\mathbf{Y} = \mathbf{W}^T\mathbf{X} \tag{3.23}$$

where $\mathbf{X}$ is the $n \times N$ data matrix, and $\mathbf{Y}$ is the $K \times N$ feature matrix, K is the dimension of the feature vectors, and $\mathbf{W}$ is the transformation matrix. Note that $K << n$, for example $K = 25$ and $n = 10,000$.

Linear Discriminant Analysis (LDA) attempts not only to reduce the dimen-

[2]Class is defined as a collection of data belonging to a particular entity, for example a collection of images belonging to a person.

sion of the data but also to maximize the difference between classes. To find the transformation $\mathbf{W}$, a generalized eigenproblem needs to be solved:

$$\mathbf{S}_b\mathbf{W} = \mathbf{S}_w\mathbf{W}\mathbf{\Lambda} \tag{3.24}$$

where $\mathbf{S}_b$ is the between-class scatter matrix and $\mathbf{S}_w$ is the within-class scatter matrix defined as:

$$\mathbf{S}_w = \sum_{i=1}^{c} P(C_i)\mathbf{S}_i \tag{3.25}$$

$$\mathbf{S}_b = \sum_{i=1}^{c} P(C_i)(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T \tag{3.26}$$

where c is the number of classes (e.g. $c = 15$), $P(Ci)$ is the probability of class i. Here, $P(Ci) = 1/c$, since all classes are equally probable. $\mathbf{S}_i$ is the class-dependent scatter matrix defined as:

$$\mathbf{S}_i = \frac{1}{N_i}\sum_{\mathbf{x}_i \in \mathbf{X}_i} (\mathbf{x}_k - \mathbf{m}_i)(\mathbf{x}_k - \mathbf{m}_i)^T \qquad i = 1, 2, \ldots, c \tag{3.27}$$

where $\mathbf{X}_i$ is the data matrix corresponding to class i.

Solving the equation 3.24, results in the following formulation which is know as Fisher's criterion [33]:

$$\mathbf{W} = \arg\max_{\mathbf{W}} \frac{|\mathbf{W}^T\mathbf{S}_b\mathbf{W}|}{|\mathbf{W}^T\mathbf{S}_w\mathbf{W}|} \tag{3.28}$$

One method for solving the generalized eigenproblem in equation 3.24 is to take the inverse of $\mathbf{S}_w$ and solve the following eigenproblem for matrix $\mathbf{S}_w^{-1}\mathbf{S}_b$:

$$\mathbf{S}_w^{-1}\mathbf{S}_b\mathbf{W} = \mathbf{W}\mathbf{\Lambda} \tag{3.29}$$

where $\mathbf{\Lambda}$ is the diagonal matrix containing the eigenvalues of $\mathbf{S}_w^{-1}\mathbf{S}_b$. But this problem is numerically unstable as it involves direct inversion of a very large matrix. One method for solving the generalized eigenvalue problem is to simultaneously diagonalize both $\mathbf{S}_w$ and $\mathbf{S}_b$:

$$\mathbf{W}^T\mathbf{S}_w\mathbf{W} = \mathbf{I} \qquad \mathbf{W}^T\mathbf{S}_b\mathbf{W} = \mathbf{\Lambda} \tag{3.30}$$

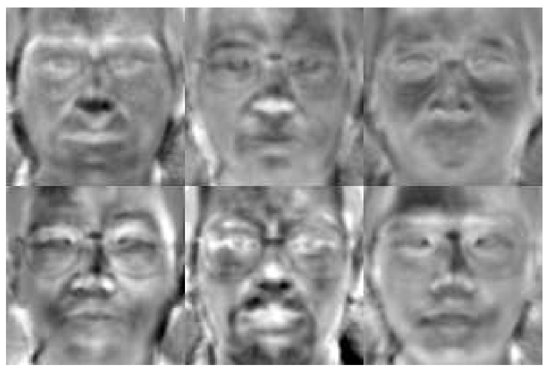

Figure 3.8: The first six LDA basis

The Direct LDA algorithm implemented in[33] is outlined below:

1. Find the eigenvectors of $\mathbf{P}_b^T\mathbf{P}_b$ corresponding to the largest K non-zero eigenvalues, $\mathbf{V}_{c\times K} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_K]$ where $\mathbf{P}_b = [\mathbf{m}_1 - \mathbf{m}, \mathbf{m}_2 - \mathbf{m}, ..., \mathbf{m}_c - \mathbf{m}]$ of size $n \times c$, $\mathbf{m}_i$ is the mean of the class i.

2. Deduce the first K most significant eigenvectors and eigenvalues of $\mathbf{S}_b$:

$$\mathbf{Y} = \mathbf{P}_b\mathbf{V} \tag{3.31}$$

$$\mathbf{D}_b = \mathbf{Y}^T\mathbf{S}_b\mathbf{Y} = (\mathbf{Y}^T\mathbf{P}_b)(\mathbf{P}_b^T\mathbf{Y}) \tag{3.32}$$

3. Let $\mathbf{Z} = \mathbf{Y}\mathbf{D}_b^{-1/2}$, which projects $\mathbf{S}_b$ and $\mathbf{S}_w$ onto a subspace spanned by $\mathbf{Z}$ this results in:

$$\mathbf{Z}^T\mathbf{S}_b\mathbf{Z} \equiv \mathbf{I} \qquad \mathit{and} \qquad \mathbf{Z}^T\mathbf{S}_w\mathbf{Z} \tag{3.33}$$

4. We then diagonalize $\mathbf{Z}^T\mathbf{S}_w\mathbf{Z}$ which is a small matrix of size $(K \times K)$

$$\mathbf{U}^T\mathbf{Z}^T\mathbf{S}_w\mathbf{Z}\mathbf{U} = \mathbf{\Lambda}_w \tag{3.34}$$

where $\mathbf{U}$ and $\mathbf{\Lambda}_w$ contains the eigenvectors and eignevalues of matrix $\mathbf{Z}^T\mathbf{S}_w\mathbf{Z}$ respectively.

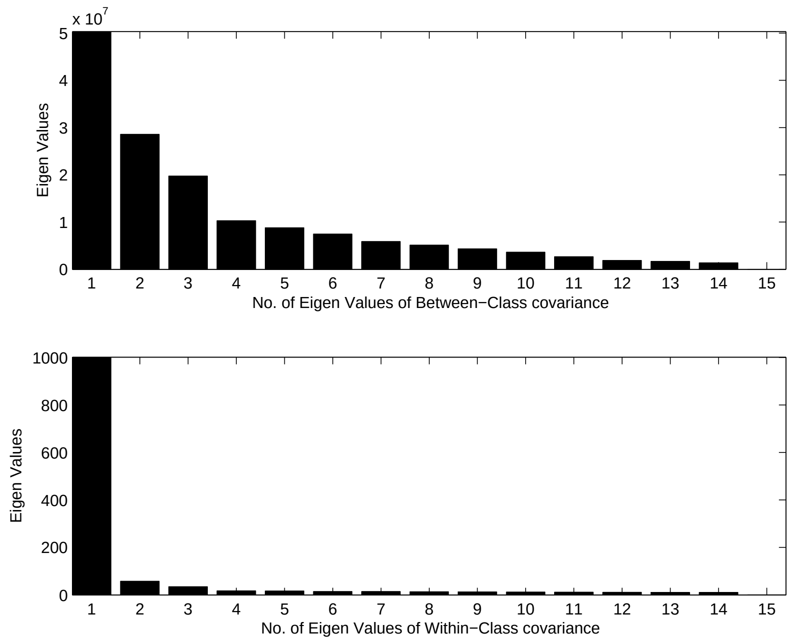

Figure 3.9: The eigenvalue spectrum of between-class and within-class covariance matrices.

5. We discard the large eigenvalues and keep the smallest r eigenvalues including the $0's$ of the matrix $\mathbf{Z}^T\mathbf{S}_w\mathbf{Z}$. The corresponding eigenvetor matrix becomes

$\mathbf{R}$ $(K \times r)$.

6. The overall LDA transformation matrix becomes $\mathbf{W} = \mathbf{ZR}$. Notice that we have diagonalized both the numerator and the denominator in the Fisher's criterion.

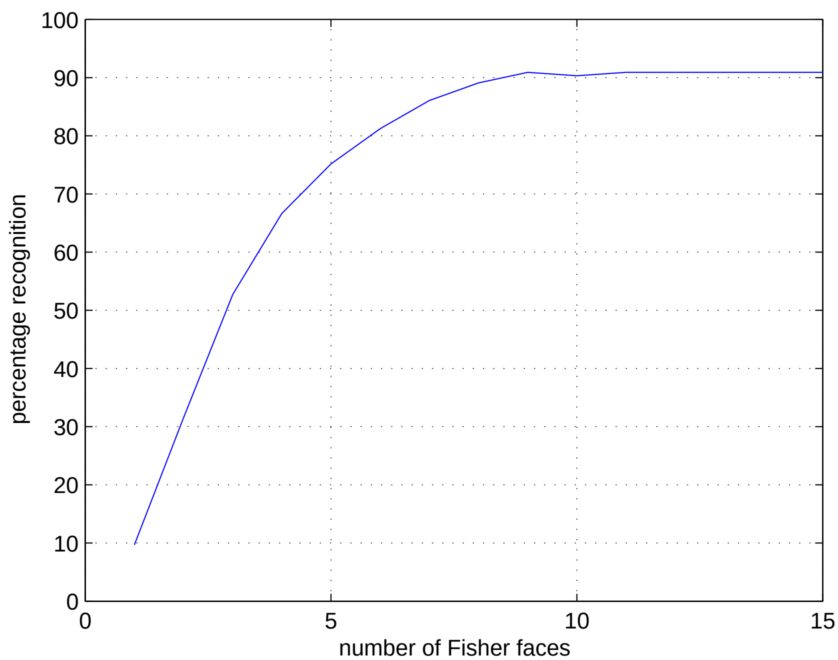

Figure 3.10: Recognition accuracy using LDA

3.3.1 Experimental Results

We have also performed a Leave-one-out experiment on the Yale faces database [11], refer to Section 3.2.1. The first six Fisher faces are shown in Figure 3.8. The eigenvalue spectrum of between-class and within-class covariance matrices is shown in Figure 3.9. It is to be noted that the eigenvalue spread for within class is very small, since we are using images of each person as elements of a class (hence lesser data spread). We notice that 13 to 15 Fisher faces are enough to reach the maximum recognition accuracy of 90.91%. The result of recognition accuracy with respect to the number of Fisher faces is shown in Figure 3.10.

3.4 Independent Component Analysis

PCA only considers second order statistics, it lacks information on the joint probability density function and higher order statistics of the elements of the transformed random vector [26]. Independent Component Analysis (ICA) accounts for such information and is used to identify independent sources from their linear combination. In face recognition, ICA is used to provide an independent rather than an uncorrelated image decomposition [26].

ICA is a statistical technique with applications in blind source separation, blind deconvolution and feature extraction. Additionally ICA is also used in the cocktail party problem, separation of artifacts in Magnetoencephalograph (MEG) data, finding hidden factors in financial data, reducing noise in natural images and Telecommunication, among others.

Independent component analysis was originally developed to deal with problems that are closely related to the cocktail party problem. In which, say two persons are speaking simultaneously and two microphones located in different positions are used to record time signal $x_1(t)$ and $x_2(t)$. The cocktail party problem is to recover the two original speech signals $s_1(t)$ and $s_2(t)$ from the recorded signals $x_1(t)$ and $x_2(t)$.

In the simplest form of ICA, one observes m scalar random variables $x_1, x_2, ..., x_m$ which are assumed to be linear combination of n unknown independent components variables, denoted by $s_1, s_2, ..., s_n$. The independent components s_i are assumed to be mutually statistically independent and zero mean. Arranging the observed variables in vector form $\mathbf{x} = (x_1, x_2, ..., x_m)^T$ and the independent component variables s_i in vector $\mathbf{s}$, the linear relationship can be expressed as:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \tag{3.35}$$

Here $\mathbf{A}$ is an unknown $m \times n$ matrix of full column rank, called the mixing matrix. The basic problem of ICA is then to estimate both the mixing matrix $\mathbf{A}$ and the observations of the independent components s_i using only observations of the mixtures x_j.

Several estimation methods for ICA have been proposed recently. The two methods most widely used in practice are the Fixed-point algorithm [46] and the Maximum Likelihood Stochastic Gradient algorithm [27]. The Fixed-point algorithm was originally derived from objective functions motivated by projection pursuit [26] and therefore its relation to estimation of the ICA components is rather indirect.

Maximum likelihood estimation is, in contrast, the mainstream method of statistical estimation.

We shall use the Fixed-point ICA algorithm to perform face recognition. The Fixed-point ICA algorithm is based on the idea rooting from the Central limit theorem. The Central limit theorem states that the sum of several independent variables, such as those in $\mathbf{s}$, tend towards a Gaussian distribution. So $x_i = a_1 s_1 + a_2 s_2 + ... + a_n s_n$ (where $i = 1, 2, ..., n$) is more Gaussian, than any single s_i. The Central limit theorem implies that if we can find a weighted sum of $x_1, x_2, ..., x_n$; with minimum Gaussianity then such weighted sum will be one of the independent components.

To proceed with this idea a measure for non-Gaussianity is needed. One such measure of non-Gaussianity is Negentropy, defined as:

$$J(y) = H(y_{gauss}) - H(y) \tag{3.36}$$

where y_{gauss} is a Gaussian random variable of the same variance as y. Where $H(y)$ is the differential entropy expressed as:

$$H(y) = -\int f(y) \log f(y) \mathrm{d}y \tag{3.37}$$

where $f(.)$ is the probability density function. In [47], an approximation of Negentropy for a random variable y is given by:

$$J(y) = c[E\{G(y)\} - E\{G(v)\}]^2 \tag{3.38}$$

where G is any nonquadratic function (e.g. $G(y) = -e^{\frac{-y^2}{2}}$), c is a constant, and v is a Gaussian random variables of zero mean and unit variance. The random variable y is assumed to be zero mean and unit variance.

In [46], equation 3.38 is used to find one independent component, by minimizing the Gaussianity of the combination of $x_1, x_2, ..., x_n$ given by $y = \mathbf{w}^T\mathbf{x}$ as follows:

$$J(\mathbf{w}^T\mathbf{x}) = c[E\{G(\mathbf{w}^T\mathbf{x})\} - E\{G(v)\}]^2 \tag{3.39}$$

Hyvarinen used this idea to derive the Fixed-point algorithm for ICA [46]. In deriving the ICA algorithm, two main assumptions are made: 1. The input components are independent, 2. The input components are non-Gaussian.

ICA for face recognition:

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N]$ denote the data matrix containing the training images in its columns. The four main stages of the ICA algorithm are: 1. Preprocessing 2. De-correlation, 3. Rotation, and 4. Projection.

1. Preprocessing: The preprocessing stage consists of centering the data matrix $\mathbf{X}$ by removing the mean vector from each of its column vectors to get new matrix $\mathbf{P}$:

$$\mathbf{P} = (\mathbf{x}_1 - \mathbf{m}, \mathbf{x}_2 - \mathbf{m}, ..., \mathbf{x}_N - \mathbf{m}) \tag{3.40}$$

2. De-correlation: The de-correlation stage consists of linearly transforming the matrix $\mathbf{P}$ such that we obtain a new matrix $\mathbf{Z}$. The matrix $\mathbf{Z}$ contains observations of a random vector $\mathbf{z}$, whose elements are uncorrelated.

$$\mathbf{Z} = \mathbf{W}_{PCA}^{T}\mathbf{P} \tag{3.41}$$

Note: The matrix $\mathbf{W}_{PCA}$ is the PCA transformation matrix, chosen with maximum number of eigen faces. Hence $\mathbf{Z}$ is a square matrix of size $N \times N$. In case of Leave-one-out experiment on Yale database, $N = 150$.

3. Rotation: The rotation stage is the heart of ICA. This stage performs source separation to find the independent components (basis face vectors). A Fixed-point ICA implementation proposed by Hyvarinen in [32] is used to find the independent components. Starting with a certain activation function $g(.)$ such as:

$$g_1(u) = tanh(a_1 u) \qquad or \qquad g_2(u) = u^3 \tag{3.42}$$

The basic iteration of the Fixed-point ICA algorithm is as follows [26]:

1. Choose $\mathbf{W}_1$ initial (e.g. random) weight

2. Let

$$\mathbf{W}_{n+1} = \mathbf{W}_n + \mathbf{W}_n[E\{\mathbf{y}g(\mathbf{y}^T)\} - diag(\beta_i)]\mathbf{D} \qquad where \quad n = 1, 2, ..., k \tag{3.43}$$

   where $\mathbf{y} = \mathbf{W}_n^T\mathbf{z}$, let $\mathbf{z}$ be the random vector whose observations are the columns of matrix $\mathbf{Z}$. The expectation operation E(.) is evaluated using the available observations. $\beta_i = E\{y_i g(y_i)\}$, where y_i is the i^th element of random vector $\mathbf{y}$. And $\mathbf{D}$ is a diagonal matrix which is expressed as $\mathbf{D} = diag(1/(\beta_i - E\{g'(y_i)\}))$.

Figure 3.11: The first six ICA basis

3. Repeat step 2 until convergence.

Note: The convergence is based on the variations of the $\beta_i s$.

4. Projection: Once the algorithm converges, we have matrix $\mathbf{W}_k$, whose size is $N \times N$, for Leave-one-out experiment on Yale database N=150. Before we can project the test and the training images onto the transformation matrix, we first obtained the overall transformation as follows (refer to section 3.2):

$$\mathbf{W}_{ICA} = \mathbf{W}_{PCA}\mathbf{W}_k \tag{3.44}$$

Hence the size of $\mathbf{W}_{ICA}$ is $n \times N$, (for Leave-one-out experiment $10,000 \times 150$). To identify test images, each image is first mean adjusted by subtracting the mean image, and then projected onto $\mathbf{W}_{ICA}$:

$$\mathbf{Y} = \mathbf{W}_{ICA}^T\mathbf{P} \tag{3.45}$$

The projection of test images are compared to those of the training images. A distance measure is used to match the test image to the training image giving the least distance in the feature space.

3.4.1 Experimental Results

We have performed a leave one out experiment on the Yale faces database [13], the same experiment as performed on LDA and PCA, refer to Section 3.2.1. The first six ICA basis vectors are also shown in Figure 3.11. The curve for recognition accuracy is shown Figure 3.12. The maximum recognition accuracy reached is 84.24%. The performance of ICA is slightly better than PCA but lesser than LDA. ICA has three main limitations: 1. Variances of the independent components can only be determined up to a scaling factor, 2. The order of the independent components cannot be determined (only determined up to a permutation order), 3. The number of separated components cannot be larger than the number of observation signals.

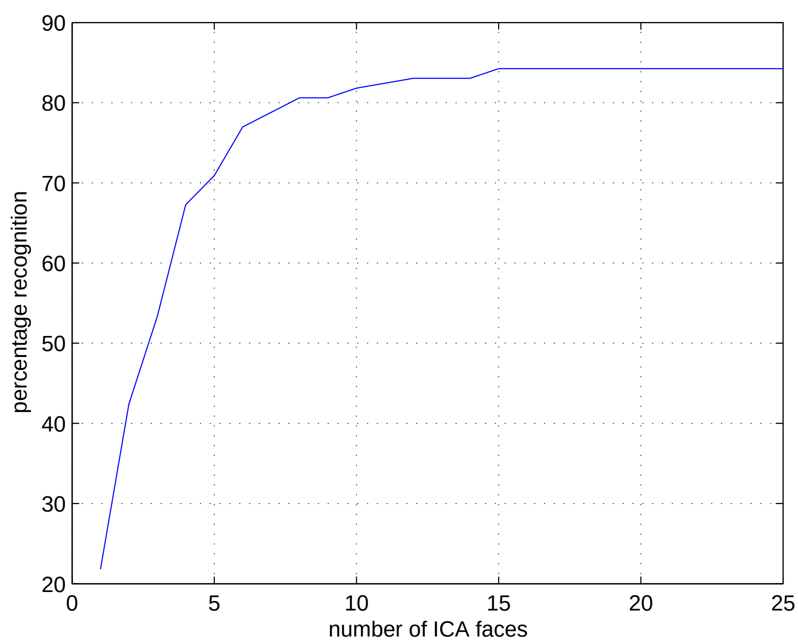

Figure 3.12: Recognition accuracy using ICA

3.5 Comparison of PCA, LDA and ICA

A comparison between PCA, LDA and ICA is given in table 3.1. The results of face recognition on the Yale database for Leave-one-out experiment are listed. It is clear from the table that LDA is superior than other algorithms with a maximum recognition accuracy of 90.909%.

It is logical that LDA performs better then PCA as LDA deals with discrimination between classes and within classes, hence efficiently utilizing class information. Whereas PCA deals with data in its entirety without paying attention to the underlying class structure. As such, the results show remarkable improvement in recognition accuracy using LDA for face recognition over PCA.

PCA only considers second order statistics. Independent component analysis concedes such information. But both PCA and ICA do not utilize class information and consider data in its entirety. There is a slight improvement in recognition accuracy in using ICA when compared to PCA. The maximum recognition accuracy of ICA is 84.4% and that of PCA is 83.3%.

The major drawbacks of ICA are that: (i) It is and iterative algorithms, hence consuming time to converge to output. (ii) The algorithm converges to different outputs on different runs (iii) It is computationally expensive when compared to PCA and LDA.

From the above discussion it is clear that LDA performs better compared to

	PCA	LDA	ICA
Accuracy	83.03%	90.91%	84.24%
Projection	Linear	Linear	Linear
Class Information	No	Yes	No
Iterative	No	No	Yes

Table 3.1: Comparison various subspace algorithms for face recognition

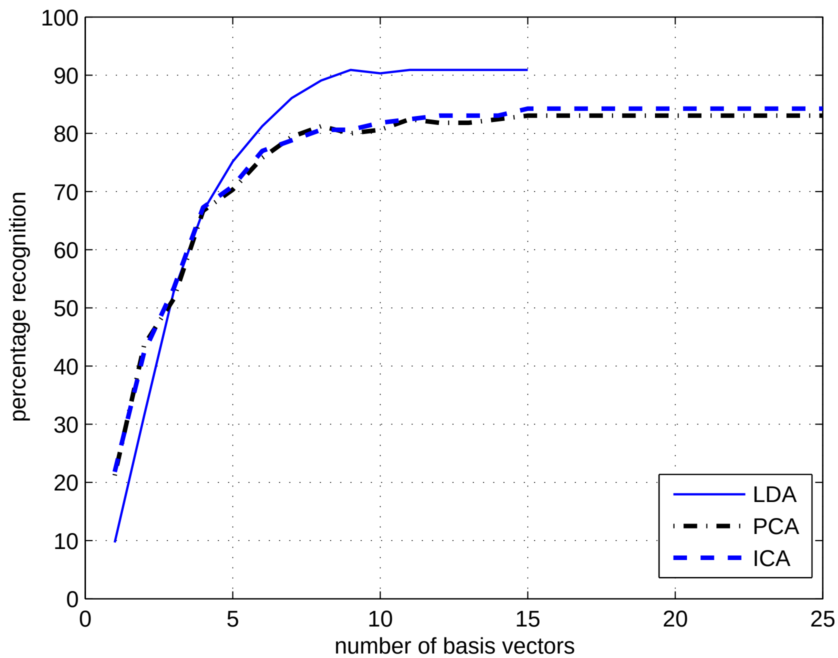

Figure 3.13: Recognition accuracy using PCA, LDA, and ICA

other algorithms. Figure 3.13 shows the recognition accuracy of PCA, LDA, and ICA. It can be noticed from the Figure 3.13 that LDA requires lesser number of Fisher faces to reach maximum recognition accuracy of 90.91 % as it uses class information. *In the figure, the curve for LDA saturates at 15 because the maximum number of Fisher faces is equal to the total number of classes, in this experiment it is limited to 15.* Based on the findings we opted in this thesis to use the LDA algorithm, particularly because it uses class information. In chapter 5, we propose a novel two stage algorithm for pose invariant face recognition using LDA, which comprises of a pose estimation stage and a view specific subspace decomposition stage.

3.6 Chapter Summary

In this chapter, we implemented a number of linear subspace techniques, including PCA, LDA, and ICA. We evaluated their recognition accuracies on the Yale database. We derived each of the 3 algorithms in detail, then we compared the various algorithms based on the leave-one-out setup experiment on the Yale database.

Chapter 4

Performance Evaluation Techniques for Face Recognition Systems

In this chapter, we present a number of experiments on face recognition. We study various distance measures including the Euclidean, Mahalanobis, City Block, etc. We also discuss two practical aspects of the problems; first what happens to the recognition accuracy when the faces are partially covered, and second, system performance on handling impostors using the Receiver Operating Characteristic (ROC) curve.

4.1 Distance Measures

Once the training stage is complete and the feature vectors corresponding to each of the training images are extracted, the system can start operating in recognition mode. In the recognition mode, the feature vector obtained from the test face is matched to each of the feature vectors corresponding to the training faces using some distance measure. The training face with the least distance is declared as match for the test image. In this section, we evaluate the performance of various distance measures with respect to recognition accuracy.

There are a number of distance measures proposed in the literature including Mahalanobis, Minkowski, Cosine metric, etc; each having unique properties. Using these distance measures, experiments were performed on AT&T database (formerly known as 'The ORL database of Faces'), [48]. The database contains ten different images of 40 distinct subjects. The benchmark subspace technique used here is PCA. We removed 3 test images per person from the database and the rest were used for training. We used the Euclidean distance as a standard to compare the recognition accuracy of various distance measures.

We arranged the feature vectors of the training images into a matrix $\mathbf{Y}$ of N $(K \times 1)$ column vectors $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N$. The distance d between a given test feature vector $\mathbf{y}_{tst}$ and each of the training feature vectors $\mathbf{y}_i$ using various distance metrics is defined as follows:

- **Euclidean distance** is a commonly used distance measure. It is basically the sum of the squared distances of two vector values $(\mathbf{y}_{tst}, \mathbf{y}_i)$. The Euclidean distance is defined by the equation [49]:

$$d^2 = (\mathbf{y}_{tst} - \mathbf{y}_i)^T(\mathbf{y}_{tst} - \mathbf{y}_i) \tag{4.1}$$

  The Euclidean distance can be seen as the shortest distance between two points, and is basically the same as Pythagoras equation when considered in 2 dimensions. The Euclidean distance is sensitive to both, adding and multiplying the vectors with some factor [50]. We have used the Euclidean distance as the standard distance for comparison purpose.

- **Standardized Euclidean distance** is defined by the following equation [49]:

$$d^2 = (\mathbf{y}_{tst} - \mathbf{y}_i)^T\mathbf{D}^{-1}(\mathbf{y}_{tst} - \mathbf{y}_i) \tag{4.2}$$

  where $\mathbf{D}$ is the diagonal matrix with diagonal elements given by v_j^2. Where v_j^2 is the variance of the j^{th} element of $\mathbf{y}$ whose observations are the $y_i's \quad (i = 1, 2, ..., N)$. The advantage of the Standardized Euclidean metric is that it takes into account the variance mentioned above. However, we found that the Standardized Euclidean distance does not improve the recognition accuracy as compared to the Euclidean distance metric (Figure 4.1). The recognition accuracy for standard Euclidean distance falls as the number of eigenfaces is increased.

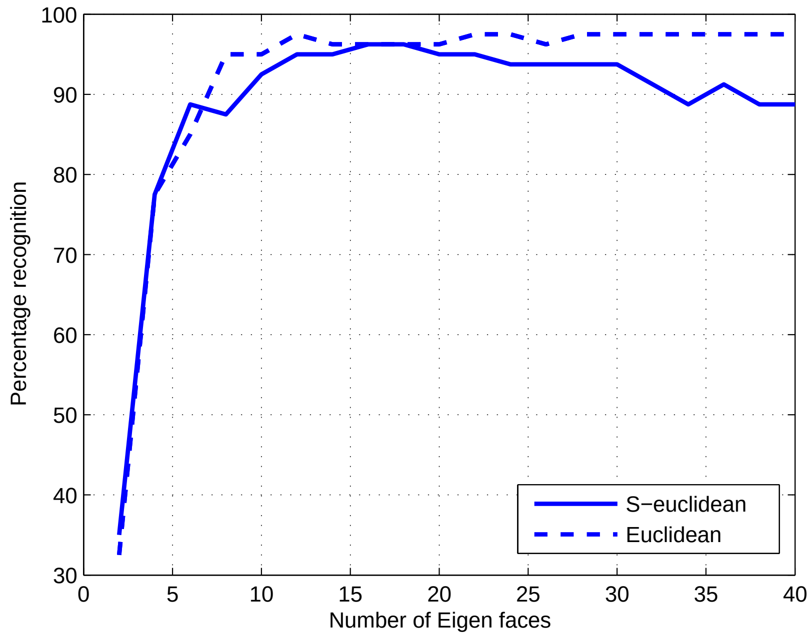

Figure 4.1: Recognition accuracy using the standard euclidean distance

- **Mahalanobis Distance** comes from the Gaussian multivariate probability density function:

$$\mathbf{p}(\mathbf{x}) = (2\pi)^{-d/2}|\mathbf{C}|^{-1/2}exp(-1/2(\mathbf{x}-\mathbf{m})^T\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})) \tag{4.3}$$

$(\mathbf{x}-\mathbf{m})^T\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})$ is called the squared Mahalanobis distance that plays an integral part in characterizing the distribution. The Mahalanobis distance is defined as [49]:

$$d^2 = (\mathbf{y}_{tst}-\mathbf{y}_i)^T\mathbf{C}^{-1}(\mathbf{y}_{tst}-\mathbf{y}_i) \tag{4.4}$$

where $\mathbf{C}$ is the estimate of the covariance matrix of $\mathbf{y}$ whose observations are the $\mathbf{y}_i s$.

We found that the Mahalanobis Distance performs poorly when compared to

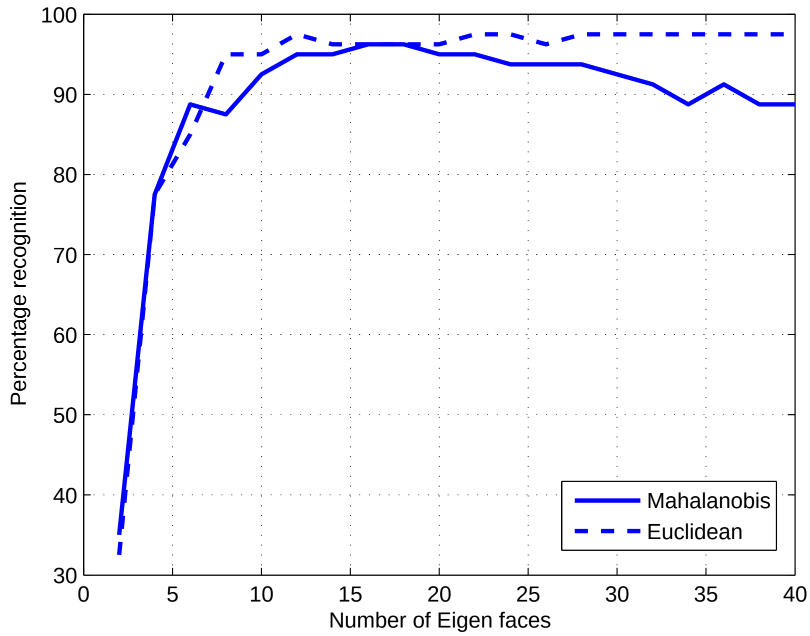

Figure 4.2: Recognition accuracy using the Mahalanobis distance

the Euclidean distance metric (Figure 4.2). We also notice that Mahalanobis distance and Standard Euclidean distance result in the same performance. The reason for this is the elements of the feature vector obtained by PCA are uncorrelated (refer to section 3.2). Hence the matrices **C** and **D** are equal.

- **City Block metric** is defined as [49]:

$$d = \sum_{j=1}^{n} |\mathbf{y}_{tstj} - \mathbf{y}_{ij}| \tag{4.5}$$

The recognition accuracy of the City Block metric is shown in Figure 4.3. City block metric also performs poorly when compared to the Euclidean distance metric.

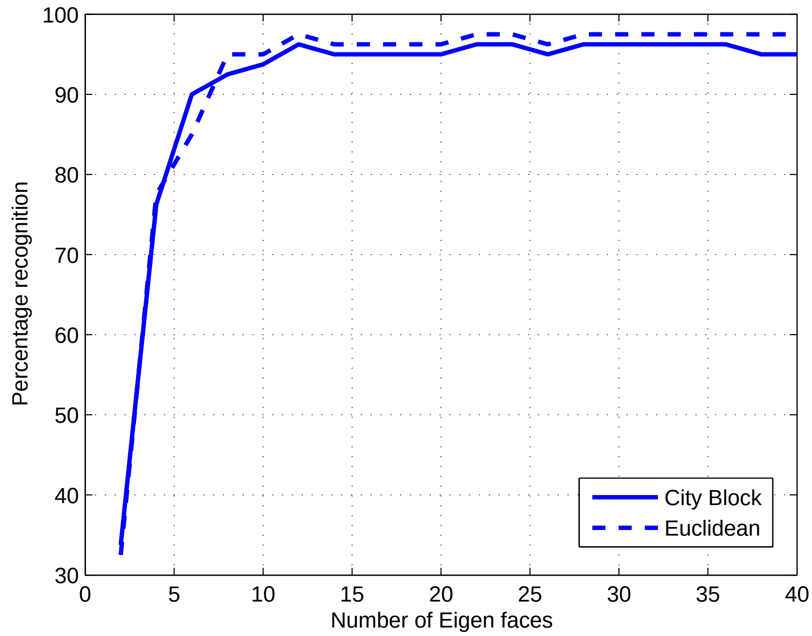

Figure 4.3: Recognition accuracy using the city block metric

- **Minkowski metric** is defined as [49]:

$$d = \{\sum_{j=1}^{n} |\mathbf{y}_{tstj} - \mathbf{y}_{ij}|^p\}^{1/p} \tag{4.6}$$

Notice that for the special case of $p = 1$, the Minkowski metric becomes the City Block metric, and for the special case of $p = 2$, the Minkowski metric gives the Euclidean distance. The resulting recognition curve for $p = 5$ is shown in Figure 4.4.

- **Cosine distance** is defined as [49]:

$$d = \{1 - \mathbf{y}_{tst}^T \mathbf{y}_i / (\mathbf{y}_{tst}^T \mathbf{y}_{tst})^{1/2} (\mathbf{y}_i' \mathbf{y}_i)^{1/2}\} \tag{4.7}$$

We found that the Cosine distance and the Euclidean distance metrics result

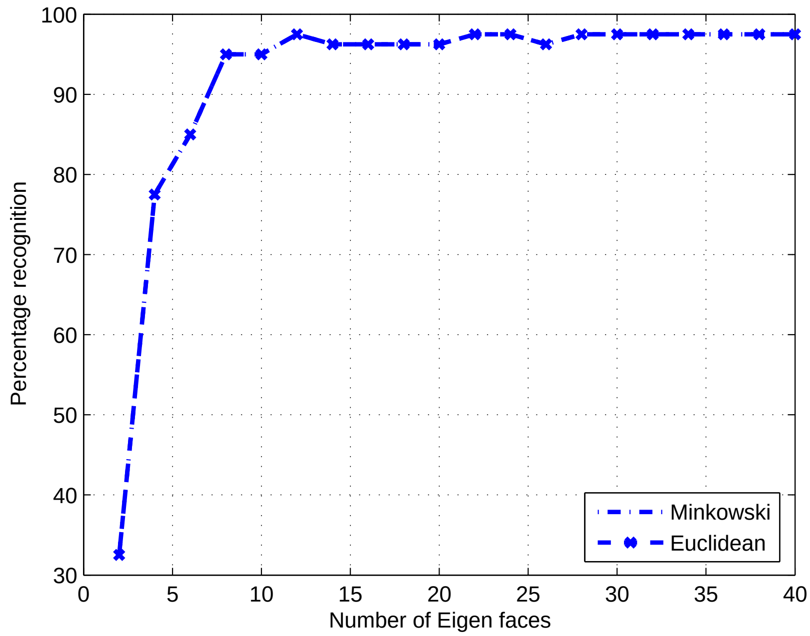

Figure 4.4: Recognition accuracy using the Minkowski metric

in comparable accuracy, refer to Figure 4.5.

Comments: From the experiments carried on various distance measures it is clear that the Euclidean distance performs better or comparable to other distance measures. For the case of Mahalanobis distance, it gives poor performance when compared to the Euclidean distance. This is because the Mahalanobis and the Standard Euclidean distances perform well with raw data, whereas in our experiments we are dealing with the distance between the linearly transformed, training and the test image vectors. In support of our results Wendy et al [50], also found that Mahalanobis distance performs poorly when compared to the Euclidean metric. Wendy et al also claim that instead of using the standard definition of Mahalanobis metric

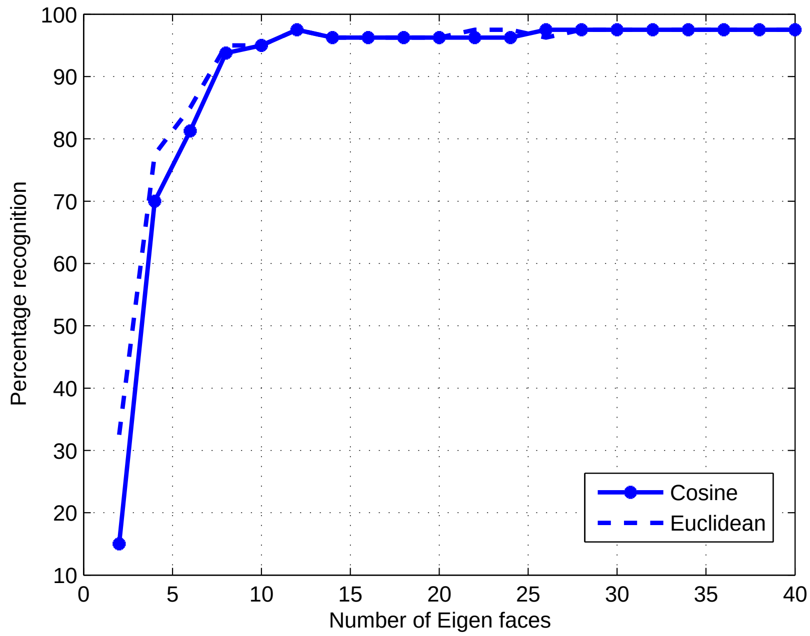

Figure 4.5: Recognition accuracy using the cosine distance

a simplified form of Moon's definition for Mahalanobis metric results in a better performance, which is defined as:

$$d(\mathbf{y}_{tst}, \mathbf{y}_i) = -\sum_{j=1}^{k} \frac{1}{\sqrt{\lambda_j}} y_{tst,j} y_{i,j} \tag{4.8}$$

where λ_j is the j^{th} eigenvalue corresponding to j^{th} eigenvector.

Considering the above, it is justified to use the Euclidean distance rather than the Mahalanobis distance metric. Additionally, since the Euclidean metric performs better than the Standard Euclidean and City block metrics; and its performance is comparable to the Minkowski and the Cosine metrics, we opted for the Euclidean distance as a standard distance measure in what follows.

4.2 Effect of Cropping

In day to day life, situations arise when the person to be recognized has his/her face covered partially with some clothing like scarf, hat, etc. It is worthwhile investigating how linear subspace techniques perform with such condition imposed. In this section, we investigate the effects of cropping on the performance of linear subspace techniques. We performed leave-one-out experiments on the Yale database using eigenfaces technique with test faces cropped.

Figure 4.6: Test faces cropped at 15%

Figure 4.7: Test faces cropped at 30%

To simulate the effects of cropping, we start by considering each of the images as a matrix of size 100×100. We then replace each element in a certain number of columns (say 15 columns) of each image by zeros to crop the face at 15%. Figure

Figure 4.8: Test faces cropped at 60%

4.6 shows test images cropped by 15%, covering the eye region and Figure 4.7 shows test faces cropped at 30% and Figure 4.8 shows test faces cropped at 60%.

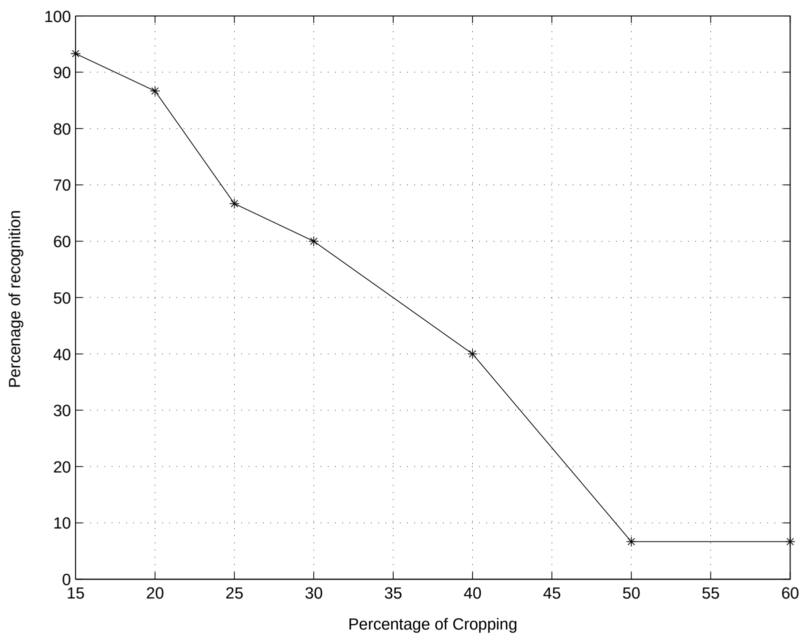

Figure 4.9: Maximum recognition accuracy at various levels of cropping

To investigate the effect of cropping, we carried out experiments on various levels of cropping ranging from 15% to 60%. The maximum accuracy at various levels of cropping is shown in Figure 4.9. We conclude that Eigenfaces algorithm performs reasonably well for cropped faces. It is noticed that for 15% cropping recognition accuracy does not degrade substantially. The recognition accuracy falls below 50% at 35% of cropping.

4.3 Face Verification and the ROC

Face recognition scenario can be classified into two types, i. Face identification or recognition and ii. Face verification. The face identification scenario is a closed test, which means all test images of an individual claiming the identity are assumed to be in the database. The system determines identity of a query image by locating the image in the database that has the highest similarity with the test image. Such a system will surely mistake an impostor, if given access to the system, to an individual in the database.

In this section we look at the face verification scenario, which is an open to universe test that compares a query face image against a template face image whose identity is being claimed. The query face image may not be in the database. To evaluate the verification performance, the verification rate (the rate at which legitimate users is granted access) vs. False acceptance rate (the rate at which an impostor is granted access) is plotted by varying a predefined threshold. The resulting curve is called The Receiver Operating Characteristic Curve (ROC). A good verification system should balance these two rates based on operational needs. Here we evaluate two types of ROC, i. The Watch List ROC and ii. The Verification ROC (defined in [51, 52]).

4.3.1 The Watch-list ROC

The watch-list ROC method of evaluation is an open-universe test. Through some method, the individual makes a claim as if he is a legitimate user from the database. Any impostor or a legitimate user can make a claim that he/she belongs to the database. In its most simple form, face recognition systems operates using a three-step process:

1. A sensor takes an observation and develops a biometric signature. The type of sensor and its observation depend on the type of biometrics device used. For face recognition, the sensor is a camera and the observation is a picture, or a series of pictures. The face recognition algorithm then attempts to find a face in the image. This can be accomplished using several methods including movement, skin tones, or blurred human shapes.

2. The second step stage of the face recognition system is to extracts the feature vectors from the face image.

3. A matching step compares the feature vector of the test image with the set (or sub-set) of feature vectors on the system's database. A measure of similarity (similarity score) or difference (distance measure) is computed for each image for the database.

The lowest distance is referred to as top match. If a distance score is lower than a preset ***threshold***, an alarm is raised. When an alarm is provided, the system thinks that the individual is located in the systems database. There are two main items of interest for watch-list applications. The first is the percentage of times the system alarms and correctly identifies (top match) a person on the watch-list. This is called the *"Detection and Identification Rate."* The second item of interest is the percentage of times the system alarms for an individual that is not on the watch-list. This is called the *"False Alarm Rate."* ([51, 52])

The False accept rate and the probability of detection and identification are not mutually exclusive. Instead, there is a give and take relationship between the two. The system parameters can be changed to receive a lower False acceptance rate by decreasing the threshold, but this also lowers the probability of verification. The Watch-list ROC is a plot of the probability of correct identification versus the probability of False alarm produced by varying the threshold.

4.3.2 Experimental Details of the Watch-list ROC

The experiments were performed using the AT&T Faces Database [48]. The data base contains ten different images of each of the 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). The "Eigenfaces algorithm for face recognition" (PCA) in [24], was used to train 30 persons $\times$ 7 faces, the remaining 3 images per person were used as test images. The 10 untrained individuals with 10 images each were used as impostors. Few training faces, the mean face and the sample of eigenfaces are shown in Figures 4.10, 4.11, 4.12 respectively.

Figure 4.10: Samples of faces from the AT&T database

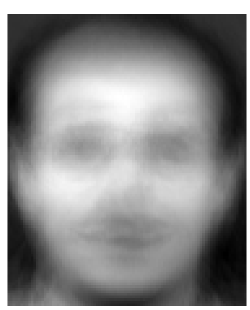

Figure 4.11: Mean face from the AT&T database

Figure 4.12: The first six eigenfaces

To evaluate the Watch-list ROC we have performed two tests on the trained system.

1. Test 1: "Test for Detection and Identification Rate": In this test, we have used 30 persons $\times$ 3 (90 test faces) faces belonging to the trained set to test detection and identification rate.

2. Test 2: "False Alarm Rate" In this test we have used 10 persons $\times$ 10 (100 test faces) untrained faces for testing False alarm rate.

The feature vectors were extracted from training images as follows:

$$\mathbf{Y} = \mathbf{W}^T\mathbf{X} \tag{4.9}$$

where $\mathbf{W}$ is PCA transformation matrix (refer to section 3.1). The feature vectors of training images can be arranged in a matrix:

$$\mathbf{Y}_{tr} = [\mathbf{y}_{tr1}, \mathbf{y}_{tr2}, ..., \mathbf{y}_{trN}] \tag{4.10}$$

where N is the number of training faces (in our case it is $7 \times 30 = 210$).

To perform the tests, the feature vectors were extracted from both sets of test faces and arranged in matrices $\mathbf{Y}_{tst}$ and $\mathbf{Y}_{imp}$. Next we evaluate the distance between each test vector in matrices $\mathbf{Y}_{tit}$ and $\mathbf{Y}_{imp}$ to the matrix $\mathbf{Y}_{tr}$ to get distance matrices:

$$\mathbf{D}_{tst} = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{N1}] \tag{4.11}$$

$$\mathbf{D}_{imp} = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{N2}] \tag{4.12}$$

where each vector $\mathbf{d}_i = [1 \times N]$, is the vector containing the distances of feature vectors of test images i with respect to feature vectors of all training faces. Before comparing the distances with some threshold value, we first scaled the distance matrices $\mathbf{D}_{tst}$ and $\mathbf{D}_{imp}$, so that the distances take a maximum value of 1. This step enables to take a threshold value between 0 and 1. Finally, in the feature space the training image with the least distance to the test image is declared as a "match" with a condition that the above mentioned distance is below the threshold value. We carry the matching process on the columns of $\mathbf{D}_{tst}$ and $\mathbf{D}_{imp}$ containing the distances of test image to the training images in feature subspace. The percent of time there is a match for the columns of $\mathbf{D}_{tst}$ is equal to percentage of detection $P(D)$ (as $\mathbf{D}_{tst}$ contains distance of legitimate users to the training images). Similarly the percentage of time there is a match for columns of $\mathbf{D}_{imp}$ is equal to False alarm P(FA). To plot the Watch-List ROC, we varied the threshold between 0.01 and 0.2.

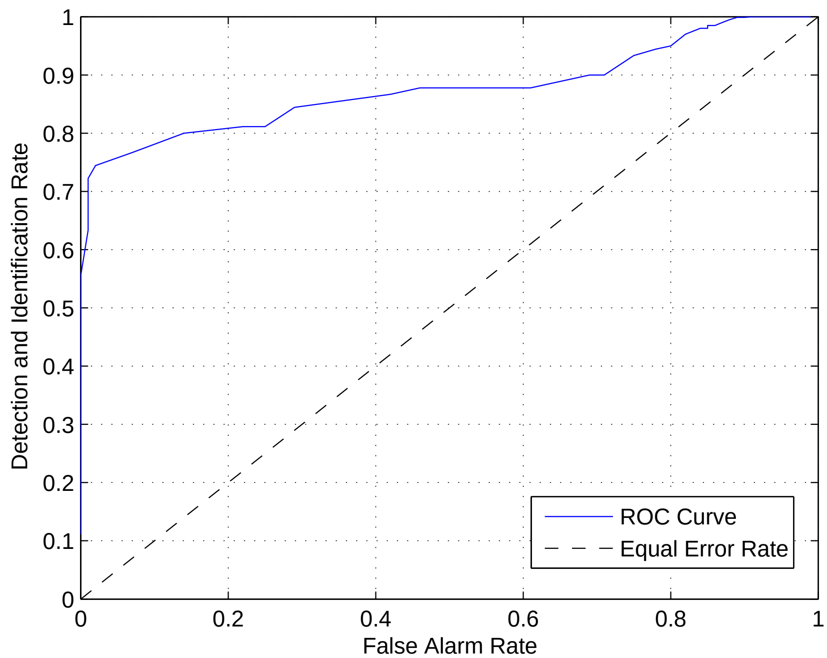

Figure 4.13: Watch-list receiver operating characteristic curve.

Performance Measure and Remarks: The performance of the ROC curve can be judged by the area under the curve, the more the area, the better is the performance. An area of 1 represents a perfect test; an area of 0.5 represents a worthless test. A rough guide for classifying the accuracy of a test is the traditional academic point system mention in [53]:

- 0.90-1 = excellent (A)

- 0.80-0.90 = good (B)
- 0.70-0.80 = fair (C)
- 0.60-0.70 = poor (D)
- 0.50-0.60 = fail (F)

The Watch-List ROC curve is shown in Figure 4.13. The area under the Watch-List ROC curve for this experiment is 0.86; hence the systems performance is considered as good. The results show that the system can be used under Watch-list mode to watch out for intruders or particular individuals among the crowd.

4.3.3 Verification ROC

The "Verification" method of using face recognition is carried out in a similar manner as the Watch-list method. The camera takes face image of an individual that may or may not be in the systems database. *Through some method, the individual makes a claim as to which feature vectors in the database is theirs.* Using PCA, a feature vector is extracted from the person's currently acquired image. *This feature vector is compared to their claimed feature vector in the system's database by using a distance measure.* If the distance score is lower than a preset threshold, the system makes decision that the individual is who they claimed to be. If the numerical value of the distance measure is greater than the preset threshold the individual is not the one who is claimed.

It is possible to have two types of errors when operating under verification task. The first occurs when the individual makes an errant claim as to their identity, but the returned distance measure is lower than the preset threshold. In this case, the system thinks that the individual is who he says he is, even though he is not. This is called the "False accept." The fraction of times that a False accept occurs across all individuals is called the "False accept rate."

The second type of error occurs when the individual makes a proper claim as to their identity, but the returned similarity score is higher than the preset threshold. In this case, the system thinks that the individual is not who he say he is, even though he really is. This is called a "False reject." The fraction of times a False reject occurs across all individuals is called the "False reject rate." Subtracting this rate from 1 (1-False reject rate) gives us the "Probability of Verification."([51, 52])

Experiments were carried on Verification ROC in the similar manner as done in Watch-list ROC, we used AT&T database. We trained 30 individuals out of 40, 7 images per person, leaving 3 out for test purpose. The ten untrained individuals with 10 images per person were used as impostors. *The major difference between "Watch-list ROC" and "Verification ROC is that in Verification ROC the individual claims to be a person already trained by the system and his current biometric signature is matched only to that individual. Where as in Watch-list ROC an individual claims to be trained by the system without stating his identity and hence his current biometric signature is matched to all individuals in the systems database to reach a decision.*

The resulting Verification Receiver Operating Curve is shown in Figure 4.14. As mentioned earlier measure of the performance of ROC curve is the area under the curve. The area under the verification ROC curve is 0.98 (close to the ideal case area=1), which shows the performance is much better than Watch-list ROC curve (area=0.86).

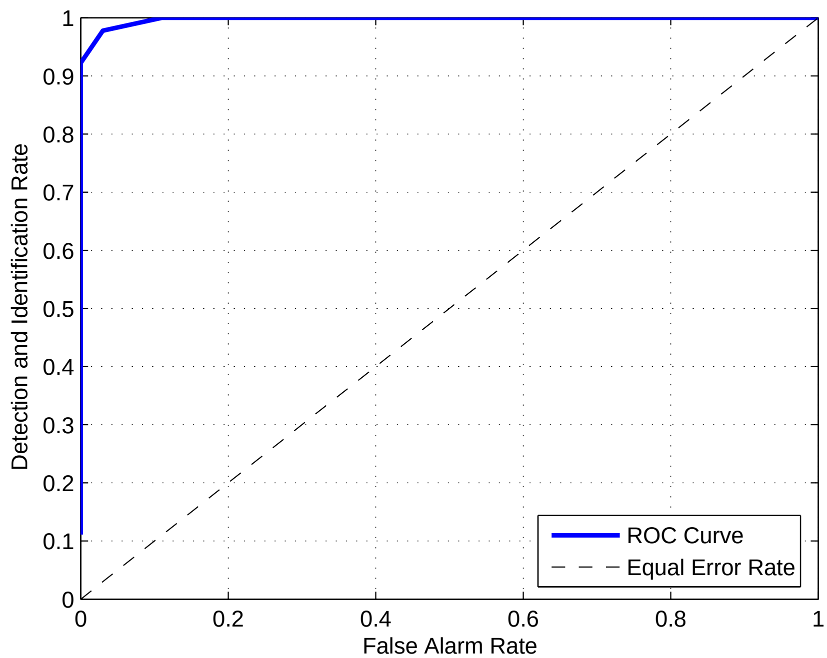

Figure 4.14: Verification receiver operating characteristic curve.

Remarks: The area under the curve for the Verification ROC indicates that it is an excellent system, but with a extra condition that user has to provide his identification. Such systems are suitable where very high security is demanded, like access to bank account, e-commerce, social security, etc.

4.4 Chapter Summary

In this chapter we first look at the various distance measures with an aim to choose the best among them. We show that Euclidean metric and Minkowski metric perform better than the others. Next, we examined the effect of cropping on recognition accuracy, we found that up to 15% cropping, there is no substantial drop in recognition accuracy. Finally we looked at a different aspect of evaluating face recognition systems performance for handling impostors using Receiver Operating Characteristic Curves (ROC). We evaluated two kinds of ROC's, the "Watch-list ROC" and the "Verification ROC." We observed that the area under the curve is 0.86 for Watch-list ROC which indicated systems performance is good. Similarly we found the area under the curve for Verification ROC which is 0.98, hence the system's performance is excellent. But with verification ROC user has to specify a user's name or some information along with a face image.

Chapter 5

Pose Invariant Face Recognition

5.1 Introduction

The face recognition problem has been studied for more than two decades. In most systems, however, the input image is assumed to be a fixed size, clear background mug shot. However, a robust face recognition system should allow flexibility in pose, lighting and expression. Face images are high dimensional data and face features have similar geometrical configuration. As such, under general conditions where pose, lighting, and expression are varying, the face recognition task becomes very complicated. The reduction of that variability through a preliminary classification step enhances the performance of face recognition systems.

Pose variation is a non trivial problem to solve, as it introduces new information with change in pose. There has been a number of techniques proposed to overcome

Figure 5.1: A typical subject in different poses

the problem of varying pose for face recognition. One of these was the application of Growing Gaussian Mixtures models [10], GGMMs are applied after reducing the data dimensions using PCA. The problem is that since Growing GMM is a probabilistic approach, it requires sufficient amount of training faces which are usually not available (example 50 faces to fit 5 GMM's). One alternative is to use a 3 dimensional model of the face [11]. However 3D models are expensive and difficult to develop. The View-based eigenspaces of Moghaddam and Pentland [12] have also shown that separate eigenspaces perform better than using a combined eigenspace for the pose-varying images. This approach essentially consists of several discrete systems (multiple observers), that is to use different subspaces for different poses. We extend this method and apply it using Linear Discriminant Analysis. In Our experiments, we will show that View-based LDA performs better than View-based

PCA. We will also demonstrate that LDA can be used to perform pose estimation.

The chapter is outlined as follows. The proposed algorithm is first described in detail. Experiments performed on the system using the UMIST database are then explained. A comparison of the proposed algorithm with the existing algorithm PCA, LDA, and View-based PCA is performed. Finally, the experiments performed on the KFUPM database for more than 3 views is explained at the end.

5.2 The Proposed Algorithm

We propose here a new system which is invariant to pose. The system consists of two stages, during the first stage, the pose of the test face is estimated using Linear Discriminant Analysis. A decision is made as to which of the 3 poses, the test face belongs. In stage two, a prior trained view specific Linear Discriminant Analysis trained for the 3 views is used to recognize the person to whom the test face belongs. The block diagram is shown in Figure 5.2. To train the three view specific LDA's, we first organize the images from the database into three different views and find the subspace transformation for each of these views. In the block diagram, we show the sizes of the matrices at different stages so as to get the notion of dimensionality reduction. The matrices $\mathbf{XL}$, $\mathbf{XR}$, and $\mathbf{XF}$ are of size 2128×60 (3 images/person, 20 persons, 2128 pixels per image). $\mathbf{WL}, \mathbf{WR}$, and $\mathbf{WF}$ are the transformation matrices; each containing K basis vectors (where $K = 20$). $\mathbf{YL}, \mathbf{YR}$, and $\mathbf{YF}$ are the transformed matrices called template matrices; each of size $K \times 60$.

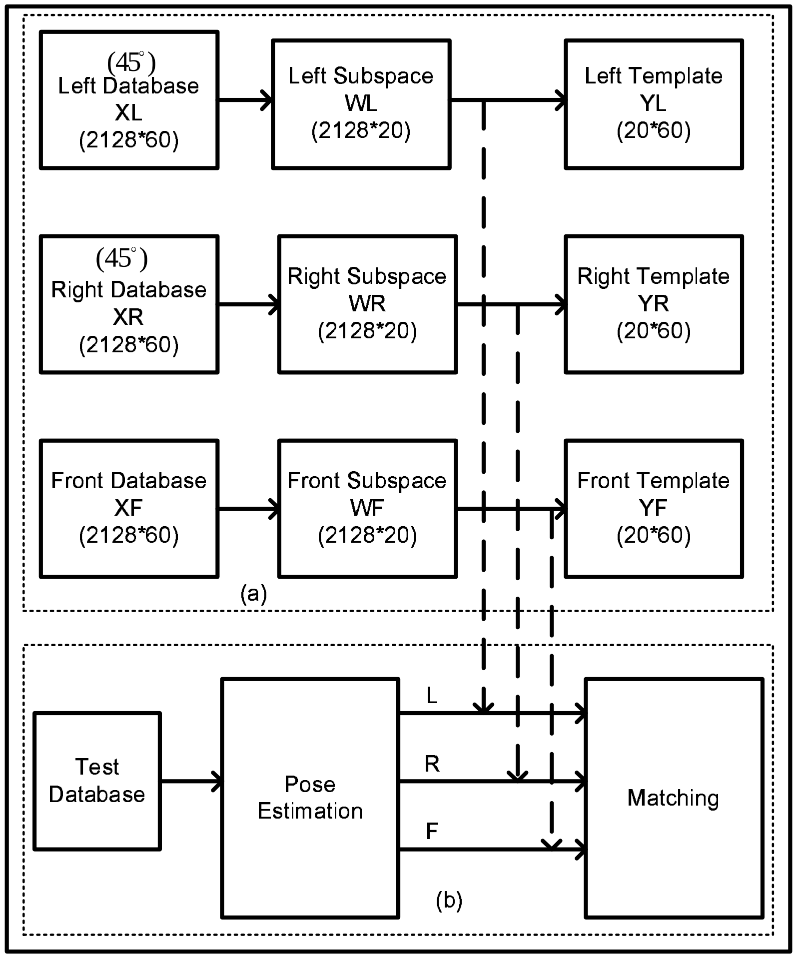

Figure 5.2: Block diagram of the pose invariant subspace system

5.3 Pose Estimation using LDA

In this system, the vector image is compared with the database of transformed images in various face poses. The pose estimation technique is composed of a *Learning stage* and a *Pose matching stage*. In this work, we consider three possible classes for the pose. These are: the left pose at 45^o (C1), the front pose (C2), and, the right pose at 135^0 (C3). Some authors considered 5 and 7 possible rotation angles for the overall pose invariant system. But experiments done on the KFUPM Capstone database show that the 3 angles mentioned above are enough to capture the main features of the face and it gives better results than system with 5 or 7 poses.

In the *Learning stage*, we first form the training database by selecting images from few subjects in 3 different poses, left pose at 45^o, the front pose, and the right pose at 135^o:

1. ***Constructing the data matrix:*** We arrange images in 3 different poses to form the pose data matrix

$$\mathbf{X}_{pose} = [\mathbf{X}_1 \in leftpose, \mathbf{X}_2 \in rightpose, \mathbf{X}_3 \in frontpose] \tag{5.1}$$

   where $\mathbf{X}_1$ is the data matrix formed by choosing images only in the left pose. Similarly $\mathbf{X}_2$ and $\mathbf{X}_3$ are data matrices formed by using images only in right and front poses respectively.

2. ***Calculating the transformation matrix:*** The first step in finding the transformation matrix is to evaluate between-class and within-class covariance matrices, (see section 3.2 for details).

In the *Pose matching* stage, we project the images from the training database onto the columns of the transformation matrix $\mathbf{W}_{pose}$ to get the feature vectors:

$$\mathbf{Y}_{pose} = \mathbf{W}_{pose}^T \mathbf{X}_{pose} \tag{5.2}$$

To estimate the pose of a given test image $\mathbf{x}_{test}$, we first mean adjust it $(\mathbf{x}_{test} - \mathbf{m})$, then project it over the columns of matrix $\mathbf{W}_{pose}$. The obtained feature vector is then compared to each of the feature vectors in $\mathbf{Y}_{pose}$. The pose corresponding to the training image with minimum distance (Euclidean sense) is taken as the pose of the test image.

Pose Estimation using PCA: Hideo [54] proposed a face pose estimating system based on eigenspace analysis. But in his approach to the problem, he used computer generated facial images in various poses to make the database. Here we use the database generated by selecting images from few subjects in 3 different poses, left pose at 45^o, the front pose, and the right pose at 135^o to form a data matrix $\mathbf{X}_{pose}$. Next, we trained the PCA algorithm using the matrix $\mathbf{X}_{pose}$ to obtain the transformation matrix $\mathbf{W}_{pose}$. To estimate the pose of a given test image $\mathbf{x}_{test}$, we first mean adjust it, $(\mathbf{x}_{test} - \mathbf{m})$, then project it over the matrix $\mathbf{W}_{pose}$. The obtained feature vector is compared to each of the feature vectors from the training database. The pose corresponding to the training image with minimum distance (Euclidean sense) is taken as the pose of the test image. The important point to note here, is that there is no class information and the matrix $\mathbf{W}_{pose}$ is evaluated in same manner as in Section 3.2 (PCA).

The advantage of using PCA and LDA algorithms for pose estimation is that the matching process is fast. Since it is performed by matching feature vectors of training database to feature vectors of test images which are small in dimensions.

The proposed pose estimation algorithm can be used to continuously track face pose of a person from a video.

5.3.1 Experimental Results of Pose Estimation

The experiments were carried on the UMIST database [55], which contains 20 persons and a total of 564 faces in varying poses. Our aim was to identify the pose of the subject so that the appropriate View-Based LDA can be used. We have performed pose estimation using both LDA and PCA. The experiments were carried using 3 poses for each of the 20 persons. We trained the system using 10 persons, and tested it using the remaining 10 persons. The results of pose estimation are summarized in Table 5.1.

Note that the LDA outperformed PCA in pose estimation. The reason being the ability of LDA to use class information while PCA only classifies features. As mentioned previously, LDA maximizes the ratio of between class covariance matrix of the projected samples to within class covariance matrix of the projected samples.

No.of test images	PCA	LDA
90	90%	100%
180	88.333%	98.888%

Table 5.1: Experimental results of pose estimation

Pose Estimation with Additive Noise

Additive noise can often hinder accurate classification of objects or face images. There are many such conditions in which noise effects the image quality, for example, variations in the detector sensitivity, environmental variations, the discrete nature of radiation, transmission or quantization errors, etc. It is very important to test the performance of pose estimation in presence of noise especially that our algorithm is a two stage algorithm, and the performance of first stage effects the performance of overall system.

Noise is described by an additive noise model, where the recorded image $f(i, j)$ is the sum of the true image $s(i, j)$ and the noise $n(i, j)$:

$$f(i, j) = s(i, j) + n(i, j) \tag{5.3}$$

The noise $n(i, j)$ is often Gaussian zero-mean and described by its variance σ_n^2. The impact of the noise on the image is often described by the signal to noise ratio (SNR), which is the ratio of signal power to noise power given by

$$SNR = \frac{Signal \quad power}{Noise \quad power} = \frac{\sigma_s^2}{\sigma_n^2} \tag{5.4}$$

where σ_s^2 and σ_n^2 are the variances of the true image and the noise, respectively. The SNR is often defined in decibels and is expressed as follows:

$$SNR_{dB} = 10 \log_{10}(SNR) = 10 \log_{10}(\frac{\sigma_s^2}{\sigma_n^2}) \tag{5.5}$$

For most applications, additive noise is assumed to be Gaussian with zero mean and variance σ_n^2 with power evenly distributed over frequency. Such noise is called additive white Gaussian Noise (AWGN).

Figure 5.3: Sample images with additive noise (SNR=10dB)

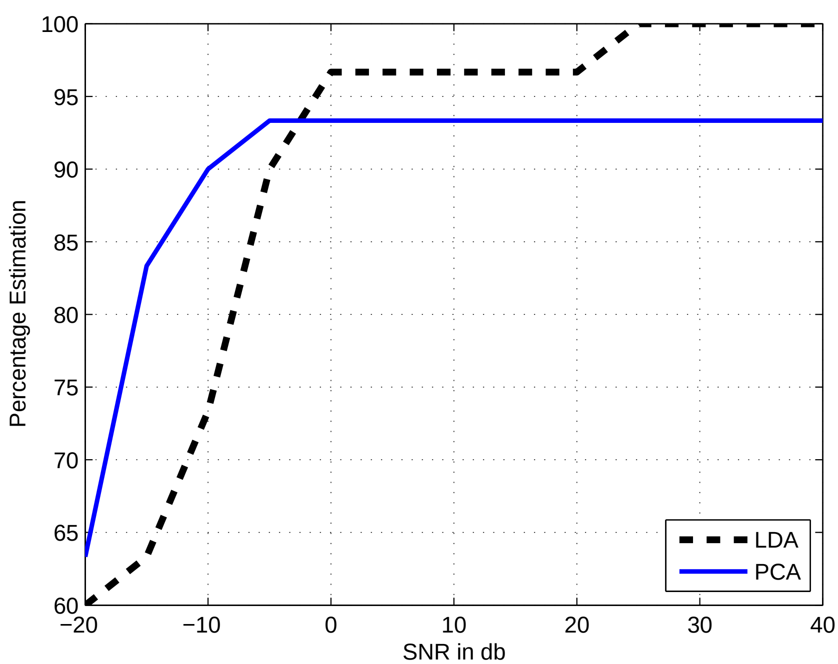

Figure 5.4: Effect of noise on pose estimation

We carried a number of experiments with artificially noisy images corrupted with additive white Gaussian noise. Some sample images are shown in Figure 5.3. The percentage recognition with respect to SNR is shown in Figure 5.4. We noticed that, for a controlled amount of SNR, the performance of pose estimation system did not degrade substantially. When the SNR falls below 0 *db*, the performance starts to deteriorate. But we still notice that the performance of LDA is better than that of the PCA.

5.4 View Specific Subspaces

Once we have estimated the pose of the subject, we now have some useful information about the test image. We utilize such information to identify the person. Since we are considering 3 poses, to setup the system, we train images of all subjects in 3 different poses left, front, and, right separately. The recognition of the test image is carried in the subspace selected for the pose of the test image. To perform recognition in a specific pose we first created 3 databases for the 3 poses.

Setting the database:

In our experiments, we used the UMIST database [55] from which 3 images per person in 3 different views (left 45^o, front, and right 135^o) were chosen, as shown in Figure 5.5. We selected the images from a particular pose (say front pose) of all subjects and arranged them in a matrix $\mathbf{XF} = (\mathbf{x}_{f1}, \mathbf{x}_{f2}, \ldots, \mathbf{x}_{fN})$, where N is the number of images in the front pose. Similarly we formed matrices **XR**, and **XL** for right 135^o and left 45^o poses respectively.

Figure 5.5: Training images of one person in different poses

5.5 Training the View Specific Databases

In this stage, we trained the pose specific matrices $\mathbf{XL}, \mathbf{XR}$, and **XF** using a linear subspace technique to derive the matrices $\mathbf{WL}, \mathbf{WR}$, and **WF**. Next we extracted the feature vectors from the matrices $\mathbf{XL}, \mathbf{XR}$, and **XF** by projecting these onto the columns of matrices $\mathbf{WL}, \mathbf{WR}$, and **WF** respectively to get matrices $\mathbf{YL}, \mathbf{YR}$, and **YF**. The following equation is an example of such projection:

$$\mathbf{YL} = \mathbf{WL}^T\mathbf{XL} \tag{5.6}$$

We performed extensive experiments on the proposed technique using the UMIST database [55] which has 20 subjects in varying poses. Few test faces of a subject are shown in Figure 5.6. We trained the view specific matrices (**XL**, **XR**, and **XF**) with 2 different algorithms; first the LDA, and the second is PCA. We shall call these *"View-based LDA"* and *"View-based PCA"*. We also trained the data with a single PCA and a single LDA and called these *"Traditional PCA"* and *"Traditional LDA"*.

Figure 5.6: Test faces of one person in 4 poses

5.5.1 View-based LDA

To build the 3 separate pose specific LDA's; we use the matrices **XL**, **XR**, and **XF**. Each of these matrices contains image samples for each of the persons in the database. The mean image of matrices **XL**, **XR**, and **XF** are shown in Figure 5.7. Using eigenvalue eigenvector decomposition of the estimated covariance matrices from **XL**, **XR**, and **XF**, we determine the LDA transformations **WL**, **WR**, and **WF**. Examples of different Fisherfaces are given in Figures 5.8, 5.9, and 5.10.

Figure 5.7: Mean images of all faces in front, left, and right pose

5.5.2 View-based PCA

Training View-based PCA is similar to the View-based LDA, where each data matrix is trained separately using PCA (refer to section 3.2 for PCA). In the case of View-based PCA there is no class information. The first four eigenfaces for each of the views are shown in Figures 5.12, 5.13, and 5.14.

5.5.3 Traditional PCA and Traditional LDA

We also carried experiments using a single PCA and a single LDA. Here, we use a single data matrix $\mathbf{X}$ which contains all poses. We train this matrix with PCA and LDA algorithms to get matrices $\mathbf{W_{PCA}}$, and $\mathbf{W_{LDA}}$. The mean image of the matrix $\mathbf{X}$ is shown in Figure 5.16. The TPCA eigenfaces are shown in Figure 5.15, while the TLDA Fisher faces are shown in Figure 5.11.

Figure 5.8: Fisher faces trained for front faces

Figure 5.9: Fisher faces trained for left faces

Figure 5.10: Fisher faces trained for right faces

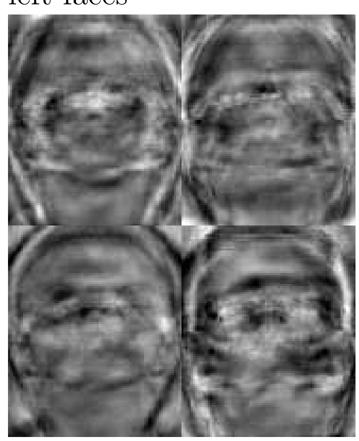

Figure 5.11: Fisher faces of traditional LDA

Figure 5.12: Eigen faces trained for front faces

Figure 5.13: Eigen faces trained for left faces

Figure 5.14: Eigen faces trained for right faces

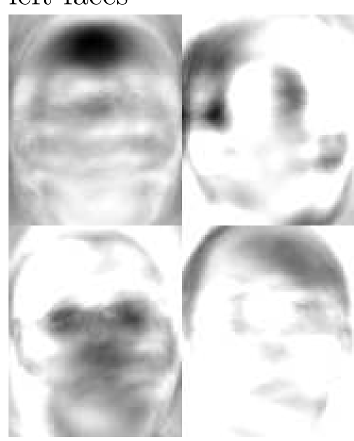

Figure 5.15: Eigen faces of traditional PCA

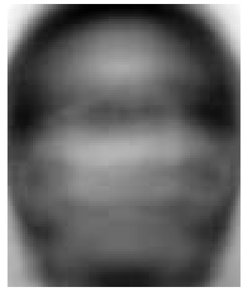

Figure 5.16: Mean images of all faces in traditional PCA and traditional LDA

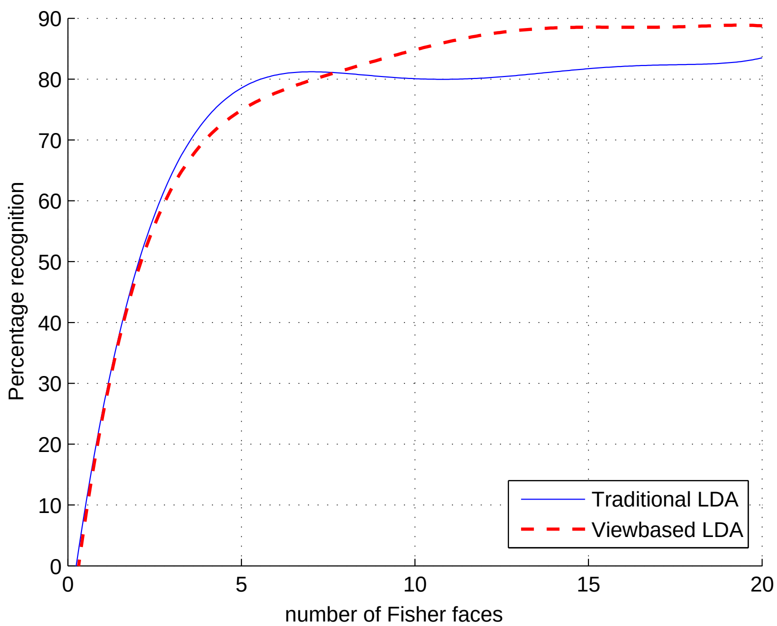

Figure 5.17: View-Based LDA vs traditional LDA

5.6 Experimental Results

We have carried our experiments on the proposed View-based LDA and compared the results to other algorithms. In the first experiment, we compared the View-based LDA to the Traditional LDA [33] (TLDA), (Figure 5.17). We noticed an improvement of 5% in recognition accuracy. The VLDA also performed better than the TPCA [24](Figure 5.18). Notice, however that the Traditional PCA performs better than View-based LDA for basis vectors less than 10. Below 10 basis vectors, relevant class information required for proper classification in the VLDA is lost, resulting in poorer performance.

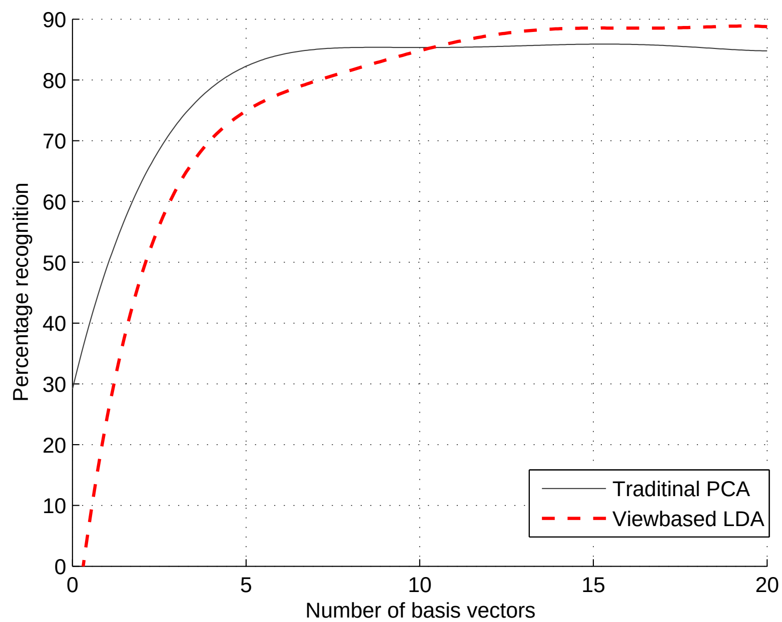

Figure 5.18: View-Based LDA vs traditional PCA

Experiments were also carried on PCA [24] and were compared to the results of View-based PCA [15]. We found that there was not much improvement in the recognition results (Figure 5.19). The reason for this could be that PCA just relies on uncorrelating data and hence training view specific PCA's do not help much

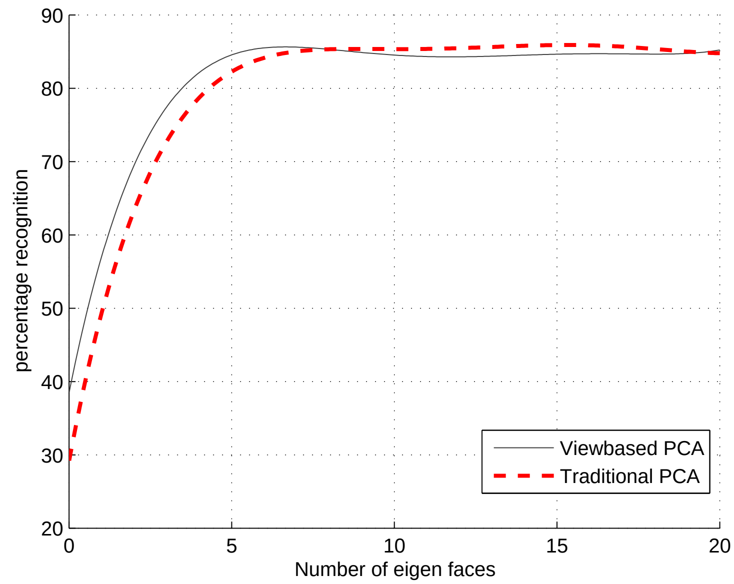

Figure 5.19: View-Based PCA vs traditional PCA

in improving the class separation. For all experiments, we see that the proposed View-Based LDA performs better than Traditional LDA and Traditional PCA.

Algorithm	Time in seconds	Max. Recognition accuracy	Memory used
View-based LDA	183.650	88.3%	514320
Traditional LDA	288.719	83.3%	429200
View-Based PCA	90.4850	84.4%	514320
Traditional PCA	140.9060	84.4%	429200

Table 5.2: Summary of results

We also noticed that the performance of LDA gets better as the number of sample images per person increases. Table 5.2 summarizes the recognition accuracy, and clearly shows that VLDA outperforms all existing algorithms, followed by VPCA, with maximum number of Fisher/eigen faces being 20. The computational complexity and memory usage of all algorithms were very comparable (see Table

5.2).

5.7 The KFUPM Capstone Database

There are a number of limitations to the UMIST database: first, it contains images of all subjects in angles varying from extreme left to front pose and not to extreme right. And secondly, the total number of images per person is around 25. These two factors were the major limitations which restricted further testing of our algorithms for 5 and 7 poses. Hence we carried further experiments using the KFUPM Capstone database.

The Capstone database of images is constructed form videos of persons turning their face form extreme left pose to extreme right pose. First, the videos are shot, then MATLAB is used to extract images form the videos. The database contains a total of 60 videos for 20 different persons. For each person, three different videos were taken. One of the three videos is stored in the training database to generate the feature matrix for the person by extracting images from it. While the other two videos are stored as test videos to extract images and test the recognition process.

There were some restrictions applied in creating the database. One of these restrictions was the homogeneous background that was used when capturing the movies of a person. Also, the light condition in the studio was adjusted to take good quality movies.

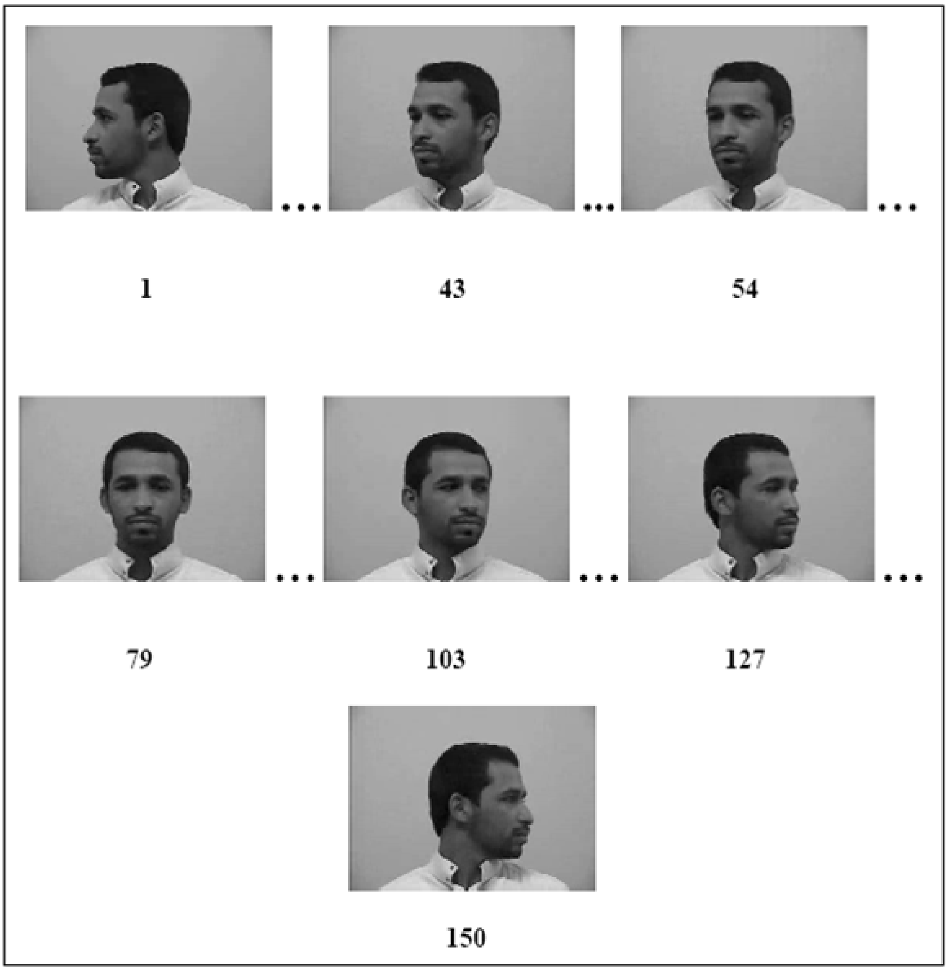

Figure 5.20: Snapshots of the 150 images generated from a video

Capturing Frames from Video:

To create a video, the camera is positioned in front of the face of a person and the person is asked to move his face horizontally from 0^o to 180^o. Each video is composed of a number of frames. All movie clips in the database were recorded for a duration of 5 seconds. Additionally, the frame rate, was adjusted to 30 frames per second. On many occasions the duration of movie clip exceeds 5 seconds, to solve this problem, the frame rate was altered to make the duration of to the movie 5 seconds. The number of frames (images) for each video was calculated as follows:

$$number \quad of \quad frames = Duration(seconds) * Frame - rate(frames/second)$$

$$= (number \quad of \quad images)$$

Since we captured the video at a rate of 30 frames per second and for a duration of 5 seconds, the number of images per video will be 150. Few sample images from the 150 images of a person are shown in Figure 5.20. These 150 images were stored in 7 different folders for the 7 different poses as shown in table 5.3. Similarly for 5 poses and 3 poses these 150 images were arranged into 5 and 3 folders respectively, as shown in tables 5.4, and 5.5.

Range of the Face Orientation	Folder's Name	Number of Images
$0^o - 15^o$	left3	13 images
$15^o - 45^o$	left2	25 images
$45^o - 75^o$	left1	25 images
$75^o - 105^o$	front	25 images
$105^o - 135^o$	right1	25 images
$135^o - 165^o$	right2	25 images
$165^o - 180^o$	right3	13 images

Table 5.3: Distribution of the captured images over the 7 poses intervals

5.8 Pose Estimation for 3, 5, and 7 pose systems

The images extracted from the video's were carefully selected to form the data matrices. For the seven poses situation, two images for each pose for each person

Range of the Face Orientation	Folder's Name	Number of Images
$0^o - 36^o$	left2	30 images
$36^o - 72^o$	left1	30 images
$72^o - 108^o$	front	30 images
$108^o - 144^o$	right1	30 images
$144^o - 180^o$	right2	30 images

Table 5.4: Distribution of the captured images over the 5 poses intervals

Range of the Face Orientation	Folder's Name	Number of Images
$0^o - 60^o$	left1	50 images
$60^o - 120^o$	front	50 images
$120^o - 180^o$	right1	50 images

Table 5.5: Distribution of the captured images over the 3 poses intervals

were used to form the data matrix. Since there were 20 persons in the database, the total number of images for each person was 14 and hence the total number of images is 280 images. For five pose situation, 2 images per view per person were used, that meant a total of 200 images. Finally, for three poses, 4 images per person per pose were used to give a total of 240 images to create the data matrix. For pose estimation, one transformation matrix was constructed for the three, five, and seven pose situations using PCA and LDA. Taking into account some practical considerations in setting the experiments, the results achieved are summarized in the Table 5.6. The seven and five pose cases gave poor results in pose estimation.

No.of poses	7	5	3
percentage pose estimation PCA	91%	94%	100%
percentage pose estimation LDA	93%	95%	100%

Table 5.6: Experimental results of pose estimation

5.9 Experiments using the View-based system for 3, 5, and 7 poses

For testing the recognition accuracy of the View-based LDA and View-based PCA for the case of 3, 5, and 7 pose situation, the data matrices were generated according to tables 5.3, 5.4, 5.5.

After estimating the pose of the test images, each image was projected on its corresponding pose specific transformation matrix to obtain the test feature vector which is then compared to the corresponding training feature vectors. The results for the experiments are summarized in table 5.7, which show recognition accuracy achieved. We notice that View-based LDA performs better then the View-based PCA in case of 3, 5 and 7 poses.

	7 poses	5 poses	3 poses
Maximum Recognition Accuracy (PCA)	93.9%	93.5%	95%
Maximum Recognition Accuracy (LDA)	94%	95%	97%

Table 5.7: Summary of experiments on recognition accuracy

Conclusions: From the experiments on the UMIST and the KFUPM Capstone database we conclude that:

1. The pose estimation accuracy falls as the number of poses is increased, that is 3 poses system gives better results for pose estimation when compared to 5 and 7 pose situations.

2. Overall, the View-based LDA gives better results than the View-base PCA.

5.10 Chapter Summary

In this chapter, we discussed the problem of pose, which is a major issue in face recognition problems. Pose variation is a non-trivial problem to solve as it involves nonlinear transformation. In this chapter, we proposed a novel approach to train view specific LDA's for pose invariant face recognition. This is achieved in two stages: first a novel pose estimation algorithm based on the LDA is used to identify the pose of the test image, and in the second stage the test image is projected onto the LDA corresponding to the specific pose. Finally, matching the obtained feature vectors to the training feature vectors is carried in the feature domain. It is found in our experiments that View-based LDA is invariant to pose and noise and achieves better recognition accuracy when compared to the Traditional LDA, the Traditional PCA and the previously developed View-based PCA. One might argue that instead of the 3 poses, the 5 and 7 poses should give better results. But the experiments performed on the KFUPM Capstone database for the 3, 5, and 7 pose system showed that the 3 pose system performed better than two others. The major reason for this behavior is that the pose estimation stage for the 5 and 7 pose system gives poor results, and as such resulting in poor recognition accuracy.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Face recognition has received considerable attention from both industry and the research community. The reason for this interest is the ease with which a face image can be acquired with minimum hardware added. However face recognition is susceptible to real life conditions like light, expression pose variation, etc. In this thesis, we investigated various linear subspace techniques like PCA, LDA and ICA. Apart from extracting the adequate feature vectors with lower dimension from the face images, these techniques led to robust face recognition algorithms but were limited in terms of variations in expression, cropping, and noise.

Each subspace technique has its own advantages and disadvantages, PCA is a technique used to uncorrelate input data. Whereas LDA utilizes class information in the input data to extract the feature vectors.

In this thesis, we opted for LDA which uses class information, we proposed a novel two stage algorithm for pose invariant face recognition, which comprises of a pose estimation stage and a view specific subspace classifier. We found that pose estimation using LDA outperforms PCA. We also found that the overall system, View-based LDA, outperforms most existing algorithms when we have varying poses. The overall aim of the thesis, as such, has been achieved with good success. Future research can be undertaken to further enhance the system performance.

6.2 Proposed Future Research Directions

Face recognition becomes a complicated task when we deal with individuals in unconstrained environments like living rooms, offices, etc. A number of requirements need to be considered when designing such systems. The first one is the "acquisition requirement", which deals with the problems of detecting and localizing a face image in non-homogeneous backgrounds. The second one is the "face recognition requirement" which is the ability to work under low resolution and other factors like light intensity, occlusion, expression variation, pose variation etc. Finally the "Identity requirement", concerned with the learning of new individuals identifying intruders, and the ability to forget known individuals.

In this thesis, our work was mainly concerned with developing a pose invariant face recognition system using a View-based LDA approach. We have shown that the proposed system can handle variations in expressions, occlusion, cropping and noise.

Some of the proposed research directions that could follow from this work are listed below:

1. ***Distance Measures:*** Here we have used the simple Euclidean distance. A number of other distance measures have been proposed in this thesis and the literature, these need to be investigated further. Another possibility is to model the distances in a Bayesian framework. Such Bayesian framework is then used in classification. An alternative to the Bayesian approach is to train the distance vectors as inputs to a Neural Network classifier.

2. ***Database Management:*** In relation to database management, we propose an extension to find efficient methods for handling users. Various aspects of managing the database include learning new users, forgetting old users and managing large databases, etc. The problem of standard database for benchmarking is still also open.

3. ***Classifier Combination:*** Here we have discussed the PCA and LDA systems separately. An extension to this work is to develop a classifier which combines the output of these two classifiers to enhance classification accuracy.

Bibliography

[1] http://www.biometricgroup.com/. International biometric group's market report 2000-2005, September 11 2001.

[2] L.C. Jain, U. Halici, I. Hayashi, S.B. Lee, and S. Tsutsui. *Intelligent Biometric Techniques in Fingureprint and Face Recogntion*. CRC Press LLC, 1990.

[3] http://www.iridiantech.com/how/index.php. Iridiantech.

[4] A. K. Jain, A. Ross, and S. Pankanti. A prototype hand geometry-based verification system. *Proc. of Audio- and Video-Based Personal Identification (AVBPA-99)*, pages 166–171, March. 1999.

[5] Mingfu Zou, Jianjun Tong, Changping Liu, and Zhengliang Lou. On-line signature verification using local shape analysis. Proceedings. Seventh International Conference on Document Analysis and Recognition, 3-6 Aug. 2003.

[6] http://library.thinkquest.org/16985/dnamain.htm. Dna.

[7] L. Hong and A. K. Jain. Integrating faces and fingerprints for personal identification. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(12):1295–1307, Dec 1998.

[8] Xiaoguang Lu. Image analysis for face recognition. Dept. of computer science and Engineering Michigan State University, East Lansing, MI, 48824.

[9] L. Wiskott, J.M. Fellous, N. Krger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE. Trans. Pattern Analysis and Machine Intelligence*, 19(7):775–779, 1997.

[10] Volker Blanz, Sami Romdhani, and Thomas Vetter. Face identification across different poses and illuminations with a 3d morphable model. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 202–207, 2002.

[11] http://cvc.yale.edu/projects/yalefaces/yalefaces.html. Yale university face database.

[12] Anil K. Jain, Arun Ross, and Salil Prabhakar. An introduction to biometric recognition. *IEEE Trans. Circuits Syst. Video Techn.*, 14(1):4–20, 2004.

[13] Volker Blanz, Sami Romdhani, and Thomas Vetter. Face identification across different poses and illuminations with a 3d morphable model. *in Proc. IEEE*

International Conference on Automatic Face and Gesture Recognition, 7(202-207), 2002.

[14] Ralph Gross, Jie Yang, and Alex Waibel. Growing gaussian mixtures models for pose invariant face recognition. 15th International Conference on Pattern Recognition, 2000.

[15] Moghaddam and A. Pentland. Face recognition using view-based and modular eigenspaces. *SPIE*, 2277:12–21, 1994.

[16] J. Daugman. Recognizing persons by their iris patterns. *BIOMETRICS: Personal Identification in Networked Society*, 1999.

[17] L. Hong, A. K. Jain, and S. Pankanti. Can multibiometrics improve performance? *AutoID'99*, pages 59–64, Oct 1999.

[18] L. I. Kuncheva, C. J. Whitaker, C. A. Shipp, and R. P. W. Duin. Independence good for combining classifiers? *International Conference on Pattern Recognition (ICPR)*, pages 168–171, 2001.

[19] Arun Ross and Anil Jain. Information fusion in biometrics. *Pattern Recognition Letters*, 24:2115–2125, 2003.

[20] http://en.wikipedia.org/wiki/. Wikipedia, the free encyclopedia, Feb. 2005.

[21] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Transactions on PAMI*, 1993.

[22] P. Rauschert, A. Kummert, M. Krips, and Y. Klimets. Face recognition by means of template matching in frequency domain - a hardware based approach. *The 2002 45th Midwest Symposium on Circuits and Systems, 2002. MWSCAS-2002*, Aug. 2002.

[23] Xiaoyan Mu, M. Artiklar, M.H. Hassoun, and P. Watta. Training algorithms for robust face recognition using a template-matching approach. *International Joint Conference on Neural Networks, 2001. Proceedings*, 3, July. 2001.

[24] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):72–86, 1991.

[25] Tsuhan Chen, Yufeng Jessie Hsu, Xiaoming Liu, and Wende Zhang. Principle component analysis and its variants for biometrics. *ECJ*, pages 61–64, 2002.

[26] Aapo Hyvrinen and Erkki Oja. Independent component analysis: A tutorial. *Helsinki University of Technology, Laboratory of Computer and Information Science*, 1999.

[27] A.J. Bell and T.J. Sejnowski. An information-maximation approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[28] C. Jutten and J. Herault. Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10, 1991.

[29] P. Comon. Independent component analysis - a new concept? *Signal Processing*, 36:287–314, 1994.

[30] A. Hyvrinen. Survey on independent component analysis. *Neural Computing Surveys*, 2:94–128, 1999.

[31] A. Hyvrinen. New approximations of differential entropy for independent component analysis and projection pursuit. *Neural Information Processing Systems*, 10:273–279, 1998.

[32] A. Hyvrinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans. on Neural Networks*, 10(3):626–634, 1999.

[33] Hua Yu and Jie Yang. A direct lda algotithm for high dimensional data with application to face recognition. *Pattern Recognition*, 34:2067–2070, september 2001.

[34] Aleix M. Martinez and Avinash C. Kak. Pca versus lda. *IEEE transaction on Pattern Analysis and Machine intelligence*, 23(2):228–233, 2001.

[35] Kwang In Kim, Keechul Jung, and Hang Joon Kim. Face recognition using kernel principal component analysis. *Signal Processing Letters, IEEE*, 9, Feb. 2002.

[36] http://www.wiau.man.ac.uk/ ct. Kernel principal component analysis.

[37] P. Phillips, H. Wechsler, J. Huang, and P. Rauss. The feret database and evaluation procedure for face-recognition algorithms, 1998.

[38] Jian Yang, A.F. Frangi, Jing-Yu Yang, David Zhang, and Zhong Jin. Kpca plus lda: a complete kernel fisher discriminant framework for feature extraction and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Feb. 2005.

[39] Zhang Yankun and Liu Chongqing. Face recognition using kernel principal component analysis and genetic algorithms. *Proceedings of the 2002 12th IEEE Workshop on Neural Networks for Signal Processing, 2002*, Sept. 2002.

[40] Xiaoguang Lu, Yunhong Wang, and A.K. Jain. Combining classifiers for face recognition. *Proceedings. 2003 International Conference on Multimedia and Expo, 2003. ICME '03*, 3, 2003.

[41] C. Basso, T. Vetter, and V. Blanz. Regularized 3d morphable models. *First IEEE International Workshop on Higher-Level Knowledge in 3D Modeling and Motion Analysis, 2003. HLK 2003*, Oct. 2003.

[42] Nazish Jamil, Samreen Iqbal, and Naveela Iqbal. Face recognition using neural networks. *Proc. of the 2001 IEEE INMIC conf.*, pages 277–281, 28-30 Dec 2001.

[43] Hazem Bouattour, Francoise Fogelman Soulie, and Emmanuel Viennet. Neural nets for human face recognition. *IEEE*, 1992.

[44] Rafael C. Gonzalez and Paul Wintz. *Digital Image Processing*. Addison-Wesley publishing company, second edition, 1987.

[45] D.N. Lawley and A.E. Maxwell. *Factor Analysis as a Statistical Method*. Butterworth, London, 1963.

[46] A. Hyvrinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997.

[47] Aapo Hyvarinen. Fast and robust fixed-point algorithm for independent component analysis. *IEEE Transactions on Neural Networks*, 3, May. 1999.

[48] http://www.uk.research.att.com/facedatabase.html. The orl database of faces.

[49] http://www.mathworks.com/access/helpdesk/help/toolbox/stats/pdist.html. The mathworks, inc: Pairwise distance between observations.

[50] Wendy S. Yambor, Bruce A. Draper, and J. Ross Beveridge. Analyzing pca-based face recognition algorithms: Eigenvector selection and distance measures. *2nd Workshop on Empirical Evaluation in Computer Vision*, 2000.

[51] Duane M. Blackburn. Face recognition 101: A brief primer. DoD counterdrug Technology Development Program Office.

[52] P.J. Phillips, P. Grother, R. Micheals, D.M. Blackburn, and E. Tabassi. Face recognition vendor test 2002. www.frvt.org.

[53] Thomas G. Tape. Interpreting diagnostic tests. Technical report, University of Nebraska Medical Center, 1999.

[54] Hideo Saito, Akihiro Watanabe, and Shinji Ozawa. Face pose estimation system based on eigen space analysis. *Proceedings. International Conference on Image Processing. ICIP 99*, 1999.

[55] Daniel B Graham and Nigel M Allinson. Characterizing virtual eigensignatures for general purpose face recognition. *Face Recognition: From Theory to Applications, NATO ASI Series F, Computer and Systems Sciences*, pages 446–456, 1998.

Vitae

- Mohammed Aleemuddin.
- Born in Hyderabad, India.
- Received Bachelor of Engineering (B.E) degree in Electronics and Communication Engineering from Osmania University, Hyderabad, India in 2001.
- Joined King Fahd University of Petroleum and Minerals in September 2002.
- Publication:

  1. Mohamed Deriche and Mohammed Aleemuddin, A Pose Invariant Face Recognition System using Linear Discriminant Analysis, *Published and presented in 2nd IEEE GCC Conference (23-25 Nov 2004 at Bahrain).*

  2. Mohamed Deriche and Mohammed Aleemuddin, Pose Invariant Face Recognition using Subspace Techniques, *Computer Aided Intelligent Recognition Techniques and Applications, (Chapter 12), Wiley Publishers.*
- Email: md.aleem@gmail.com